Scanned by 4osh

AMIGA

Amiga DOS 1.3
Benutzerhandbuch
Deutsch

Taken from Amiga-Manuals-Website

Commodore

URHEBERRECHT

Commodore behält sich vor, die Software, die Namen der Disketten, aber auch die Zusammenstellung der Disketteninhalte den aktuellen Notwendigkeiten anzupassen, so daß der im vorliegenden Handbuch beschriebene Stand nicht als Maßstab für Liefer- und Leistungsumfang genommen werden kann.

Der Inhalt dieses Handbuchs kann ohne Ankündigung geändert werden und ist nicht als eine Garantieerklärung anzusehen.

Alle Rechte vorbehalten, einschließlich derer, dieses Buch oder Teile daraus in irgendeiner Form ohne schriftliche Genehmigung von Commodore zu vervielfältigen.

© Copyright der deutschen Ausgabe bei Commodore Büromaschinen GmbH, Frankfurt 1988

Part.Nr. 363077-01

Die folgenden Zeichensätze sind im Zeichensatzverzeichnis auf der Extras-Diskette enthalten und urheberrechtlich geschützt (c) 1985, 1987 Adobe Systems, Inc.:

Times Roman Helvetica Medium Courier

4 89 1 2 3 4 90 B

WARENZEICHEN

Amiga ist ein eingetragenes Warenzeichen der Commodore-Amiga Inc., Amiga 500, Amiga 2000, AmigaDOS, Amiga Workbench und Amiga Kickstart sind Warenzeichen der Commodore-Amiga Inc. Commodore, das Commodore-Firmensymbol und CBM sind eingetragene Warenzeichen der Commodore Electronics Ltd.

Alphacom ist ein eingetragenes Warenzeichen der Alphacom Inc.
Alphapro ist ein Warenzeichen der Alphacom Inc.
Brother ist ein eingetragenes Warenzeichen der Brother Industries Ltd.
Canon ist ein eingetragenes Warenzeichen der Canon USA Inc.
CityDesk ist ein eingetragenes Warenzeichen der MicroSearch Inc.
ColorMaster ist ein Warenzeichen von CalComp.
Diablo und Xerox sind eingetragene Warenzeichen der Xerox Corporation.
Epson ist ein eingetragenes Warenzeichen der Epson America Inc.
IBM ist ein eingetragenes Warenzeichen der International Business Machines Corp.
Imagewriter ist ein Warenzeichen der Apple Computer Inc.
LaserJet, LaserJet PLUS und PaintJet sind eingetragene Warenzeichen der Hewlett-Packard Company.
Microsoft ist ein eingetragenes Warenzeichen der Microsoft Corp.
NEC und Pinwriter sind eingetragene Warenzeichen von NEC Information Systems.
Okidata ist ein eingetragenes Warenzeichen von Okidata, einer Abteilung der Oki America Inc.
Okimate 20 ist ein Warenzeichen von Okidata, einer Abteilung der Oki America Inc.
PageSetter ist ein eingetragenes Warenzeichen der Gold Disk Inc.
Proprinter XL ist ein Warenzeichen der International Business Machines Corp.
ProWrite ist ein Warenzeichen von New Horizons Software.
Publisher 1000 ist ein Warenzeichen von Brown-Wagh Publishing.
Qume ist ein eingetragenes Warenzeichen der Qume Corporation.
LetterPro 20 ist ein Warenzeichen der Qume Corporation.
Tektronix ist ein eingetragenes Warenzeichen der Tektronix Inc.

Dieses Dokument kann auch Verweise auf andere Warenzeichen enthalten, von denen angenommen wird, daß sie Eigentum der dort angegebenen zugehörigen Quellen sind.

URHEBERVERWEIS

FastFonts (FF) wird mit Erlaubnis von Charlie Heath, MicroSmith Inc. verwendet.
Besonderen Dank an Matt Dillon für das Pipe-Steuerprogramm.
BIX ist der elektronische Informationsaustausch des Byte-Magazins. Byte wird von McGraw-Hill herausgegeben.

AmigaDOS-Enhancer-Software

Dieses Enhancer-Paket enthält alle erforderlichen Softwareprodukte, um Ihren Amiga-Computer mit der neuen Betriebssystem-Version 1.3 auszustatten. Es beinhaltet:

- Kickstart-Diskette, Version 1.3 (für Benutzer des Amiga 1000)
- Workbench-Diskette, Version 1.3
- Extras-Diskette, Version 1.3
- Programm-Lizenzvereinbarung

Durch den Ausbau zur Version 1.3 wird die Leistungsstärke Ihres Amiga auf das Maximum erhöht. Die Version 1.3 verbessert die vorhandene Technologie und erweitert die Fähigkeiten des Amiga und vieler Anwendungsprogramme.

Für die volle Ausnutzung der Software-Version 1.3 ist die Verwendung der neuen Kickstart-Version 1.3 erforderlich. Wenn Sie mit einem Amiga 1000 arbeiten, ist dies extrem einfach: Verwenden Sie für den Neustart des Systems die neue Kickstart-Diskette 1.3 anstelle der alten Kickstart-Diskette 1.2.

DIE KICKSTART-DISKETTE KANN NICHT MIT DEM AMIGA 500 ODER DEM AMIGA 2000 VERWENDET WERDEN. Stattdessen kann das Kickstart-ROM in diesen Computern durch das Kickstart-ROM der Version 1.3 ersetzt werden. Das Kickstart-ROM der Version 1.3 ist bei autorisierten Commodore-Servicehändlern erhältlich. Der ROM-Baustein wird dort installiert. Bitten Sie um den Einbau des Bausteins mit der Teilenummer 363077-01.

Die neue Software kann aber auch verwendet werden, ohne daß das neue Kickstart-ROM 1.3 installiert ist. Die einzige Anwendung, die dann nicht verfügbar ist, ist die Neustart-Funktion der resetfesten RAM-Disk, die später in diesem Handbuch beschrieben ist. Außerdem kann bei Systemen des Typs Amiga 2000, die mit einer Festplattensteuerplatine A2090A ausgestattet sind, kein automatischer Neustart von der Festplatte ausgeführt werden, solange das Kickstart-ROM 1.3 nicht installiert ist.

Inhaltsverzeichnis

Kapitel 1. Der Arbeitstisch, Workbench-Diskette 1-1
1.1 SHELL, NEWCON und CLI 1-1
1.1.1 Startdateien 1-1
1.1.2 Programm-Merkmale (TOOL TYPES) 1-2
1.1.3 Mehrfache Piktogramme 1-3
1.1.4 Zusätzliche Leistungsmerkmale der SHELL 1-4
1.2 Die Systemschublade 1-7
1.2.1 NoFastMem 1-8
1.2.2 FastMemFirst 1-8
1.2.3 MergeMem 1-8
1.2.4 Format 1-9
1.2.5 SetMap 1-10
1.2.6 FixFonts 1-10
1.2.7 InitPrinter 1-10
1.3 Die Schublade Prefs 1-11
1.3.1 Das Hauptfenster des Voreinstellerprogramms Preferences. 1-11
1.3.2 Das Fenster Change Printer 1-12
1.3.3 Das Piktogramm CopyPrefs 1-16
1.4 Die Hilfsprogrammschublade (Utilities) 1-16
1.4.1 ClockPtr 1-17
1.4.2 CMD 1-17
1.4.3 InstallPrinter 1-19
1.4.4 MORE 1-19
1.4.5 PrintFiles 1-21
1.4.6 GraphicDump 1-22
1.4.7 Die Uhr (Clock) 1-23
1.5 Die beiden RAM-Disks 1-25
1.5.1 RAM-Handler 1-25
1.5.2 Die resetfeste RAM-Disk 1-26

Kapitel 2. AmigaDOS und das Verzeichnis C der Workbench-Diskette 2-1
2.1 Unveränderte Befehle 2-2
2.2 Neue oder geänderte Befehle 2-2
2.3 Formate der unveränderten Befehle 2-43

Kapitel 3. Weitere Workbench-Verzeichnisse 3-1
3.1 Geräte (Devices) 3-1
3.1.1 Liste der Geräte, die angemeldet werden können (MountList) 3-2
3.1.2 Zwischenspeicher (Clipboard.device) 3-4
3.2 Tastaturbelegungen (Keymaps) 3-5
3.3 Zeichensätze (Fonts) 3-6
3.4 Handler (Steuerprogramme) 3-7
3.4.1 Aux-Handler 3-7
3.4.2 Newcon-Handler 3-8
3.4.3 Pipe-Handler 3-9
3.4.4 Speak-Handler 3-11
3.5 SHELL-SEG 3-12
3.6 Befehlsdateien (Scripts) 3-13
3.7 Bibliotheken (Libraries) 3-14

Kapitel 4. Druckertreiber 4-1
4.1 Grafikdruck 4-2
4.1.1 Geschwindigkeit 4-3
4.1.2 Farbdruck 4-3
4.2 Unterstützte Drucker 4-4

Kapitel 5. Ändern der Startsquenzen 5-1
5.1 Öffnen einer Anfangs-SHELL 5-3
5.2 Festlegen zusätzlicher Pfade und logischer Namen 5-4
5.3 Anmelden von Festplatten-Partitionen 5-4
5.4 RAM: umbenennen 5-5
5.5 Weitere Befehle resident laden 5-5
5.6 Umgebungsvariablen festlegen 5-6
5.7 Eigene Befehlsmakros in die Datei S: SHELL-STARTUP aufnehmen 5-6
5.8 Anlegen einer resetfesten RAM-Disk (RAD) 5-6
5.8.1 Inhalt der Workbench-Diskette in die resetfeste RAM-Disk - Für Amiga-Systeme mit mehr als 2 MByte RAM 5-7
5.8.2 Kleinste resetfeste RAM-Disk für Übergabe der Steuerung an eine Festplattensteuerung A2090 und das Festplattenlaufwerk 5-8
5.8.3 Kleine resetfeste RAM-Disk für Amiga-Systeme mit 1 MByte RAM 5-12

Kapitel 6. Hilfsprogramme auf der Extras-Diskette 6-1
6.1 Programme 6-1
6.1.1 FED 6-1
6.1.2 FreeMap 6-6
6.1.3 IconEd 6-6
6.1.4 IconMerge 6-6
6.1.5 KeyToy2000 6-8
6.1.6 Palette 6-9
6.1.7 PerfMon 6-10

Kapitel 7. MEMACS 7-1
7.1 Festlegungen zur Schreibweise und spezielle Begriffe 7-1
7.2 Öffnen von MEMACS 7-4
7.3 MEMACS-Menüs 7-6
7.4 Das Menü Project 7-6
7.5 Das Menü Edit 7-11
7.6 Das Menü Window 7-16
7.7 Das Menü Move 7-17
7.8 Das Menü Line 7-19
7.9 Das Menü Word 7-21
7.10 Das Menü Search 7-22
7.11 Das Menü Extras 7-24
7.12 Befehle, die nicht in den Menüs vorhanden sind 7-30
7.13 Hinzufügen von Befehlen zur MEMACS-Startdatei 7-31
7.14 Nach Funktion sortierte Zusammenfassung der Befehle 7-32
7.15 Alphabetisch sortierte Zusammenfassung der Befehle 7-36

Anhang A. Für Amiga 2000 mit Festplattenlaufwerk A-1
A.1 Die Steuerplatine A2090A A-1
A.2 Das schnelle Dateisystem (FastFileSystem) A-1

Anhang B. Liste der Verzeichnisse/Dateien auf der Workbench-Diskette B-1

Anhang C. Liste der Verzeichnisse/Dateien auf der Extras-Diskette C-1

ÜBERSICHT UND EINLEITUNG

Willkommen bei der neuen Version 1.3, der AmigaDOS-Enhancer-Software. In diesem Handbuch, in dem vorausgestzt wird, daß Sie mit den Informationen in den Benutzerhandbüchern zu dem Amiga-Systemen vertraut sind, werden die Workbench-Version 1.3 (Kapitel 1 bis 5) und die Extras-Version 1.3 (Kapitel 6 und 7) beschrieben.

In den ersten Kapiteln werden alle Unterschiede beschrieben, die zur Basisversion der Workbench-Diskette bestehen. In dieser neuen, aktualisierten Version der Workbench-Diskette sind viele bereits bei der Version 1.2 vorhandene Merkmale verbessert. Es wurden auch einige neue Programme hinzugefügt, die Ihnen noch mehr Freude am Arbeiten mit dem System schenken sollen:

SHELL — Arbeitet wie der Befehlsprozessor CLI, erlaubt jedoch das Edieren von Zeilen und läßt Sie vorhergegangene Befehle zurückverfolgen.

FixFonts —Ermöglicht Ihnen nach dem Hinzufügen neuer Zeichensätze die Aktualisierung des Zeichensatzverzeichnisses auf der Workbench-Diskette.

MORE — Erlaubt die Anzeige von ASCII-Textdateien

CMD — Leitet die serielle oder parallele Ausgabe in eine Datei um.

Es gibt mehrere neue AmigaDOS-Befehle, und die Leistung einiger der ursprünglich vorhandenen wurde erweitert und verbessert. Außerdem wurde dem Voreinstellerprogramm Preferences ein Bildschirm für Grafikauswahl hinzugefügt, der zusätzliche Möglichkeiten beim Grafikdruck unterstützt.

Diese neuen Merkmale sind alle in diesem Handbuch beschrieben. Es führt Sie zuerst von Schublade zu Schublade, um alle Änderungen der verschiedenen Programme zu erklären. Anschließend werden für erfahrene AmigaDOS-Benutzer die neuen Befehle und Steuerprogramme behandelt und Hinweise für das Ändern der Startsequenzen gegeben.

Der letzte Teil des Handbuchs enthält Informationen über die Extras-Diskette. Im Extras-Fenster werden Piktogramme für AmigaBasic, die Schubladen BasicDemos, FD1.3, Tools und PCUtil angezeigt. Die Diskette enthält jedoch auch Verzeichnisse, die Druckertreiber, Tastaturbelegungen und Zeichensätze enthalten, die auf der Workbench-Diskette keinen Platz fanden.

Alle Hilfsprogramme in den Schubladen Tools und PCUtil werden in den Kapiteln 6 und 7 behandelt. AmigaBasic, BasicDemos und FD1.3 beziehen sich jedoch auf die Amiga-Programmiersprache Basic, die eine Anpassung des BASIC von Mikrosoft® darstellt, welche speziell für den Amiga entwickelt wurde. Die vollständigen Informationen über Amiga-Basic sind im Amiga-Basic-Handbuch zu finden, das zusammen mit Ihrem Computer geliefert wurde.

1. Der Arbeitstisch, Workbench-Diskette

1.1 SHELL, NEWCON: und CLI

Einer der deutlichsten Unterschiede zwischen den Workbench-Versionen 1.2 und 1.3 ist die Erweiterung des Workbench-Fensters um das Programm SHELL. (Beachten Sie, daß die Schublade Demos entfernt wurde.) SHELL wie auch CLI (Befehlsprozessor, der sich in der Systemschublade befindet) ermöglichen die Kommunikation mit Ihrem Amiga unter Verwendung von AmigaDOS-Befehlen. Alle Programme auf der Workbench-Diskette können von der SHELL oder dem CLI aus gestartet werden. Der Hauptunterschied zwischen den beiden ist, daß die SHELL eine erweiterte CLI-Umgebung bietet. Das bedeutet, daß die SHELL die volle Leistung des CLI und mehr besitzt.

Da SHELL und CLI viele Gemeinsamkeiten aufweisen, werden die Funktionen, die sich auf beide beziehen, zuerst erklärt. Danach werden die zusätzlichen Möglichkeiten der SHELL beschrieben.

1.1.1 Startdateien

Wird die SHELL oder der CLI durch Doppelklicken auf das entsprechende Piktogramm geöffnet, führt das Hilfsprogramm die Befehle der jeweiligen Startdatei aus. Für die SHELL ist dies die Datei S:Shell-Startup, für den CLI die Datei S:CLI-Startup. Beide Startdateien befinden sich im Verzeichnis S: der Workbench-Diskette.

Ein in beiden Startdateien verwendeter Befehl ist der Befehl PROMPT. Durch den Befehl PROMPT wird die Eingabeaufforderung festgelegt, die beim Öffnen des SHELL- bzw. CLI-Fensters erscheint. Wird der CLI geöffnet, erscheint eine einfache numerische Anzeige — 1>. Beim Öffnen der SHELL erscheint jedoch die Anzeige "1.SYS:>", die sich ändert, wenn Wechsel des aktuellen Verzeichnisses anzuzeigen sind. Wenn Sie z. B.

```
cd utilities
```

eingeben, ändert sich die Eingabeaufforderung zu:

```
1. Workbench1.3D:utilities>
```

Durch den Befehl PROMPT können die Eingabeaufforderungen der SHELL verändert werden:

PROMPT %N>	Zeigt nur die Nummer des Befehlsprozessors an
PROMPT %N.%S>	Zeigt die Nummer des Befehlsprozessors, einen Punkt und dann das aktuelle Verzeichnis an.
PROMPT%S.%N>	Zeigt das aktuelle Verzeichnis, einen Punkt und dann die Nummer des Befehlsprozessors an.

Soll beim Öffnen des Fensters eine andere Eingabeaufforderung erscheinen, edieren Sie die Datei S:Shell-Startup. Verwenden Sie dazu den AmigaDOS-Befehl ED, das Hilfsprogramm MEMACS oder einen anderen Texteditor.

1.1.2 Programm-Merkmale (TOOL TYPES)

Wenn Sie ein Piktogramm und dann INFO aus dem Workbench-Menü wählen, erscheint ein Fenster, das nähere Informationen über das Programm enthält, welches durch das Piktogramm dargestellt wird. Sie können verschiedene Variablen festlegen, die die SHELL und den CLI beeinflussen, indem Sie INFO aus dem Workbench-Menü verwenden, um die TOOL TYPES des Programms zu verändern. Die TOOL TYPES eines Programms sind eine Gruppe von Merkmalen, die das Verhalten des Programms bestimmen. Wenn Sie das INFO-Fenster öffnen, können Sie die Standardfestlegungen der TOOL TYPES unter Verwendung der Symbole ADD/DELETE im Feld TOOL TYPES ändern.

Zu den Piktogrammen für SHELL wie auch CLI können die folgenden TOOL TYPES hinzugefügt werden:

WINDOW = Festlegungen bezüglich des Fensters

Diese Option erlaubt Ihnen, die Plazierung, die Größe und den Handler zu bestimmen, den die neue SHELL oder der neue CLI verwendet. Die Festlegungen für das Fenster müssen vollständig angegeben werden, zum Beispiel:

WINDOW=CON:0/10/640/100/NEWCLI

WINDOW=NEWCON:0/1/640/100/AmigaShell

Das Format für eine Fensterfestlegung lautet: "Handler:x/y/w/h/Name", wobei gilt:

Handler: = CON: oder NEWCON:
x = <Anzahl der Bildpunkte zwischen der linken Bildschirm- und der linken Fensterkante>/
y = <Anzahl der Bildpunkte zwischen der oberen Bildschirm- und der oberen Fensterkante>/
w = <Breite des Fenster, in Bildpunkten>/
h = <Höhe des Fenster, in Bildpunkten>/
Name = <Titel des Fensters>

Im ersten oben angegebenen Beispiel öffnet sich ein CON:-Fenster bündig mit der linken Bildschirmkante und zehn Bildpunkte von der oberen Bildschirmkante entfernt. Das Fenster ist 640 Bildpunkte breit und 100 Bildpunkte hoch. In der Titelleiste des Fensters erscheint NEWCLI.

Die Handler, zwischen denen Sie wählen können, sind CON: oder NEWCON:. CON: ist das Standard-Amiga-Fenster, während NEWCON: für eine neue Alternative steht. NEWCON: erlaubt das Edieren von Befehlszeilen und verfügt über einen Puffer, der vorausgegangene Befehle bewahrt. Die SHELL ist ein Beispiel für ein NEWCON:-Fenster. (NEWCON: wird in Kapitel 3, Andere Workbench-Verzeichnisse, erklärt.)

STACK=

Diese Festlegung erlaubt Ihnen, die Anfangskapazität des durch den CL! oder die SHELL verwendeten Stack-Puffers anzugeben. Die Größe des Stack-Puffers gibt den Speicherbereich an, der für ein spezielles Programm reserviert wird. (Der Standardwert für die meisten Programme ist 4.000 Bytes.)

1.1.3 Mehrfache Piktogramme

Für beide Programme können mehrere Piktogramme erzeugt werden, indem Sie ein beliebiges Piktogramm von TYPE Project kopieren, wie z. B. die SHELL oder sogar das Piktogramm POINTER im Fenster der Voreinstellerschublade. (Sie können anhand des Eintrags TYPE im INFO-Fenster des Piktogramms feststellen, ob es sich um ein Projekt-Piktogramm handelt.) Stellen Sie das Standardprogramm im INFO-Fenster auf SYS:System/CLI ein und legen Sie für die TOOL TYPES WINDOW= und STACK= unterschiedliche Werte fest. Dies erlaubt auf einfache Weise, die Fenster an verschiedenen Positionen erscheinen zu lassen.

Wenn Sie zum Beispiel eine zusätzliche SHELL, genannt Shell2, in der Schublade Utilities erstellen wollen:

1) Verwenden Sie in CLI oder SHELL den Befehl COPY, um eine Kopie des SHELL-Piktogramms zu erstellen*.
   `COPY Sys:Shell.info to Sys:Utilities/Shell2.info`
2) Öffnen Sie das Fenster Utilities.
3) Prüfen Sie die TOOL TYPES des Piktogramms SHELL2 und stellen Sie sicher, daß als Standardprogramm (DEFAULT TOOL) SYS:System/CLI angegeben ist.
4) Ändern Sie, falls gewünscht, die TOOL TYPES WINDOW= und STACK=.

* Sie können das Piktogramm auch unter Verwendung der Funktion Duplicate im Workbench-Menü kopieren.

1.1.4 Zusätzliche Leistungsmerkmale der SHELL

Einer der Vorteile bei der Arbeit mit SHELL ist, daß viel Schreibarbeit gespart wird. Mit SHELL kann eine Befehlszeile ediert werden. Unterläuft bei Verwendung des CLI ein Schreibfehler bei der Befehlseingabe, muß die Rücktaste verwendet werden, um alle Zeichen bis zum Fehler zu löschen. Bei Verwendung der SHELL kann der Cursor dagegen mit der Cursor-Steuertaste nach links bis zum Fehler bewegt, dieser korrigiert und dann RETURN gedrückt werden. Es ist nicht erforderlich, den gesamten Befehl neu einzugeben.

Die SHELL verfügt über ein "Gedächtnis". Das bedeutet, daß zuvor eingegebene Befehle durch die Taste "Cursor aufwärts" erneut angezeigt werden können. Anstelle eine Befehlszeile, die drei oder vier Befehle zuvor verwendet wurde, erneut zu schreiben, kann jeder vorhergegangene Befehl durch Drücken der Taste "Cursor aufwärts" wieder abgerufen werden. Es kann auch durch die Eingabe eines Teils der Zeichenfolge und Drücken der beiden Tasten Shift und "Cursor aufwärts" (oder Control-R) ein bestimmter Befehl gesucht werden. So können Sie z. B. die letzte Befehlszeile suchen, durch die der Inhalt eines beliebigen Verzeichnisses angezeigt wurde, indem Sie DIR eingeben und die Tasten Shift und "Cursor aufwärts" gemeinsam drücken.

Nach dem Zurückblättern zu vorangegangenen Befehle kann durch die Taste "Cursor abwärts" wieder vorwärts geblättert werden. Dadurch läßt sich zurückverfolgen, welche Befehle durchgeführt wurden, um einen bestimmten Ablauf oder ein Resultat zu erzeugen. Das gleichzeitige Drücken der Tasten Shift und

"Cursor abwärts" (oder Control-B) bringt Sie zum Ende des Befehlspuffers und zeigt eine leere Befehlszeile an.

Einige weitere Möglichkeiten, die Befehlszeile zu bearbeiten, sind:

Die Rücktaste (BACKSPACE)	Löscht Zeichen links vom Cursor.
Die Del-Taste	Löscht Zeichen an der Cursor-Position.
Control-K	Löscht alle Zeichen von der Cursorposition bis zum Ende der Zeile.
Control-U	Löscht alle Zeichen von der Cursorposition bis zum Anfang der Zeile.
Control-X	Löscht die gesamte Zeile.
Shift-Cursor nach links (oder Ctrl-A)	Stellt den Cursor an den Zeilenanfang.
Shift-Cursor nach rechts (oder Ctrl-Z)	Stellt den Cursor an das Zeilenende.
Control-W	Bewegt den Cursor zur nächsten Tabulatorposition.

Mit der SHELL können AmigaDOS-Befehle zu Makros zusammengefaßt werden. Der Name eines Makros ist als Kurzform für häufig verwendete Befehle zu verstehen. Das Format für das Erstellen eines Makros lautet:

ALIAS <Makro-Name> <Zeichenfolge>

Wenn Sie zum Beispiel häufig den Inhalt von df1: anzeigen lassen, können Sie ein Makro mit dem Namen "d1" erstellen, das dieselbe Funktion durchführt:

```
ALIAS d1 "DIR df1:"
```

Wenn die SHELL einen Makro-Namen erkennt,

1) ersetzt sie den Makro-Namen durch die <Zeichenfolge>,
2) fügt den Rest der von Ihnen eingegebenen Befehlszeile hinzu
3) und führt die gesamte Befehlsfolge aus.

Wenn Sie Dateinamen oder andere Anweisungen innerhalb eines Makros jeweils unterschiedlich angeben wollen, können Sie eckige Klammern ([]) verwenden, um anzugeben, wo die Einfügung erfolgen soll. Zum Beispiel:

ALIAS ls "LIST [ ] NODATES"

In diesem Fall stellen die Klammern einen Dateinamen oder ein Verzeichnis dar, das anzugeben ist. Zum Beispiel wird

```
ls df1:
```

die Verzeichnisse und Dateien auf der Diskette in df1: ohne Datums- und Zeitangaben auflisten. Die Eingabe von "ls c" bewirkt eine Liste aller Dateien im Verzeichnis C ohne Datums- und Zeitangaben.

Einige spezielle Merkmale bezüglich der Makros:

- Makros sind an eine bestimmte SHELL gebunden. Makros, die in einer SHELL definiert wurden, arbeiten nicht mit einer neu erzeugten SHELL. Die Datei S:SHELL-STARTUP wird bei jeder Verwendung des Befehl NEWSHELL automatisch ausgeführt. Häufig verwendete Makros können deshalb in der Datei S:SHELL-STARTUP definiert werden, so daß sie nicht bei jedem Öffnen einer SHELL neu erstellt werden müssen (siehe Kapitel 5, Ändern der Startsequenzen).
- Die alleinige Eingabe des Befehl ALIAS bewirkt, daß alle momentan vorhandenen Makros aufgelistet werden.
- Ein Makro wird gelöscht, indem nur ALIAS <Makro-Name> ohne Zeichenfolge eingegeben wird.
- Am Ende der Zeichenfolge wird nicht automatisch ein Leerzeichen eingefügt. Wenn Sie dort ein Leerzeichen benötigen, müssen Sie es am Ende der Zeichenfolge eingeben.

Die SHELL unterstützt auch die Verwendung des neuen Befehls-Schutzbit. Bei der Version 1.2 waren die für eine Datei verfügbaren Schutzbits r (readable - lesbar), w (writeable - schreibbar), e (executeable - ausführbar) und d (deleteable - löschbar). Die Version 1.3 unterstützt einige weitere Schutzbits - Script, Archive und Pure. (Die Bits Archive und Pure sind in Kapitel 2 beschrieben. Siehe Erklärungen zur den Befehlen PROTECT und RESIDENT).

Verwenden Sie den Befehl PROTECT, um das Script-Bit für eine Datei zu setzen. Die SHELL wird dann diese Datei wie eine Befehlsdatei vom Typ EXECUTE ausführen, statt sie wie einen Befehl zu behandeln.

Für die schnellere Ausführung einer Befehlsdatei kann der Befehl EXECUTE resident in den Speicher geladen werden, indem der AmigaDOS-Befehl RESIDENT verwendet wird. Dies wird gewöhnlich in der Startsequenz durchgeführt. Für die effektivste Nutzung dieser Funktion können Sie folgendes tun:

- Stellen Sie durch den Befehl PATH sicher, daß sich das Verzeichnis S: im Befehlssuchpfad des Amiga befindet (Sie können dies in der Datei Startup-Sequence festlegen) und/oder
- Kopieren Sie das gesamte Verzeichnis S: oder häufig benutzte Befehlsdateien nach RAM: und stellen Sie sicher, daß RAM: im Befehlssuchpfad enthalten ist.

Im Verzeichnis S: sind einige Befehlsdateien enthalten, die zusammen mit dem Befehl ALIAS der SHELL verwendet werden können, um Befehlen, die diese Fähigkeit noch nicht besitzen, die Funktion mit Jokerzeichen zu ermöglichen. Diese Befehle sind SPAT, das die Prüfung übereinstimmender Muster bei Befehlen mit einem Argument erlaubt, und DPAT, das für Befehle mit doppeltem Argument verwendet wird. Zum Beispiel:

```
alias ren "execute s:dpat rename []"
alias pro "execute s:spat protect []"
```

Diese Anweisungen erlauben beim Befehl RENAME bzw. PROTECT mit Jokerzeichen zu arbeiten. (Anweisungen für die Arbeit mit Jokerzeichen können Sie dem AmigaDOS-Handbuch entnehmen. Beispiele für SPAT und DPAT sind im Kapitel 3 zu finden.)

Die SHELL unterstützt die normale Umleitung unter Verwendung der Symbole < und >, ebenso wie die Umleitung in Form des Anhängens durch >>. Umleitungen beziehen sich auf den Weg einer Eingabe für oder einer Ausgabe von einem Befehl zu einer Datei. Beachten Sie, daß >> keine Datei erzeugt, sondern lediglich Informationen an eine bestehende Datei anfügt. Es können auch Kommentare in die Befehlszeile eingegeben werden, indem ihnen ein Semikolon (;) vorangestellt wird.

1.2 Die Systemschublade

Wie bei der Version 1.2 enthält die Systemschublade den Befehlsprozessor CLI und die Programme NoFastMem, DiskCopy, FastMemFirst, Format, SetMap und InitPrinter. Es wurden jedoch zwei neue Hilfsprogramme, FixFonts und MergeMem, hinzugefügt, während das Programm Say in der Schublade Utilities und das Programm IconEd auf der Extras-Diskette untergebracht wurden.

1.2.1 NoFastMem

Einige sehr alte Softwareprogramme laufen evtl. nicht korrekt ab, wenn sich ein anderer Speicher als CHIP RAM im Amiga befindet, wie zum Beispiel FAST RAM im Amiga 2000 oder der zusätzliche RAM-Speicher, der durch eine Speichererweiterung A501 zum Amiga 500 hinzufügt wird. In diesem Fall zwingt das Doppelklicken auf das Piktogramm NoFastMem den Amiga dazu, nur den residenten Speicher CHIP RAM zu verwenden. Das Piktogramm arbeitet wie ein Umschalter. Soll die Speichererweiterung wieder vom System genutzt werden, klicken Sie erneut auf das Piktogramm NoFastMem.

Wenn Sie mit dem Befehlsprozessor (CLI oder SHELL) arbeiten, kann die Benützung der Speichererweiterung wieder aktiviert werden, indem eine Unterbrechung an das Programm NoFastMem gesendet wird. Dies erfolgt entweder über den Befehl BREAK oder das Drücken von Control-C. Control-C wirkt nur dann, wenn das Programm nicht durch den Befehl RUN gestartet wurde.

1.2.2 FastMemFirst

Dieses Programm bewirkt eine "Neuordung" der Speicherliste des Amiga, die sich dann auswirkt, wenn Speicher bei der Adresse $C00000 zugewiesen wird.

Dieses Programm erlaubt den Einsatz schnellen Speichers vor dem $C00000-Speicher, was die Arbeitsgeschwindigkeit des Systems erhöht. Sofern nicht ausdrücklich die Verwendung des CHIP RAM angefordert wird, verwendet der Amiga standardgemäß $C0000-RAM, FAST RAM und dann CHIP RAM. Durch FastMemFirst wird die Reihenfolge der Zuweisung zu FAST RAM, $C0000-RAM und dann CHIP RAM geändert.

1.2.3 MergeMem

Enthält Ihr Amiga zusätzliche RAM-Erweiterungsplatinen, kann das Programm MergeMem hilfreich sein. Wenn Sie auf dieses Piktogramm doppelklicken, unternimmt MergeMem den Versuch, die Speicherlisten (MemLists) der sequentiell konfigurierten RAM-Platinen zu mischen. Die RAM-Platinen müssen dieselben Attribute aufweisen wie angrenzende Erweiterungs-RAM-Bereiche. Speicher von getrennten RAM-Platinen wird normalerweise in separaten Speicherbereichen gehalten. MergeMem versucht, die einzelnen getrennten Speicherbereiche zu einem großen Bereich zusammenzufassen, um die Zuweisung größerer Speicherblöcke

durch Programme zu ermöglichen. MergeMem mischt nur Speicher mit identischen Attributen.

Soll MergeMem vom Befehlsprozessor aus gestartet werden, ist lediglich MERGEMEM einzugeben. Wenn es nicht möglich ist, die Speicherbereiche zusammenzufassen, erscheint eine Meldung, die die RAM-Konfiguration auflistet und angibt, daß das Mischen nicht möglich ist.

1.2.4 Format

Das Doppelklicken auf dieses Piktogramm bewirkt die Anzeige eines Kommunikationsfensters, das Sie auffordert, Initialize aus dem Menü Disk des Arbeitstischs zu wählen. Wenn Sie eine Diskette formatieren wollen, wählen Sie Initialize, oder verwenden Sie den Befehl FORMAT, wenn Sie mit dem Befehlsprozessor vertraut sind. Das Format dieses Befehls lautet:

FORMAT DRIVE <Laufwerk> NAME <Name> [NOICONS]
[QUICK] [FFS] [NOFFS] [INHIBIT]

Die Option NOICONS verhindert, daß der neu formatierten Diskette ein Piktogramm für den Papierkorb hinzugefügt wird.

Die Option QUICK gibt an, daß der Befehl FORMAT nur die allererste Spur (mit Boot-Block) und die Spur mit dem Inhaltsverzeichnis (Root-Block) formatieren und die Sektorbelegungstabelle erstellen soll. Dies ist nützlich, wenn eine bereits formatierte Diskette erneut schnell formatiert werden soll.

Die Optionen FFS und NOFFS beziehen sich auf die Festplatte. Diese Optionen heben die MountList-Schlüsselworte auf. Die Option FFS kennzeichnet die Platte für die Verwendung des FastFileSystem. Die Option NOFFS kennzeichnet die Platte für die Verwendung des Standard-Dateisystems. Da FORMAT prüft, ob ein anderes Dateisystem in MountList festgelegt ist, sind beide Optionen verfügbar. Wurde FastFileSystem angegeben, wird die Option FFS automatisch verwendet. Ist in MountList ein anderes Dateisystem festgelegt, wird die Option NOFFS eingesetzt. Falls Ihre Festplatte in zwei Partitionen aufgeteilt ist, müssen Sie das Schlüsselwort DosType in MountList entsprechend dem zu verwendenden Dateisystem festlegen (in Kapitel 3 ist ein Abschnitt über MountList zu finden).

1.2.5 SetMap

Dieses Programm erlaubt Ihnen, die für Ihre Tastatur korrekte Tastenbelegung auszuwählen. Die Tastaturbelegung gibt dem Computer an, welche Zeichen er bei Betätigung der einzelnen Tasten auf der Tastatur zu erkennen hat.

In Deutschland ist d zu wählen: SETMAP d .

Auf der Extras-Diskette (im Verzeichniss devs/keymaps) sind weitere Tastaturtreiber zu finden, die zur Workbench-Diskette kopiert werden können (siehe Abschnitt Tastaturbelegungen in Kapitel 3). Die Tastaturbelegung usa0 ist für die Kompatibilität mit einigen alten Softwareprodukten erforderlich, die voraussetzen, daß mit der Kickstart-Version 1.0 gearbeitet wird. Die Tastaturbelegung usa2 ist für die Dvorak-Tastatur gedacht. Die anderen sind internationale Belegungen für europäische und skandinavische Tastaturen.

1.2.6 FixFonts

Aktualisiert die Dateien .font im Verzeichnis FONTS:. Dieses Programm wird verwendet, nachdem Zeichensätze zu Unterverzeichnissen des Verzeichnisses FONTS: hinzugefügt oder daraus gelöscht wurden. FixFonts korrigiert alle Dateien .font, so daß sie den aktuellen Inhalt der Unterverzeichnisse in FONTS: exakt wiedergeben.

1.2.7 InitPrinter

Die Auswahl dieses Piktogramms initialisiert Ihren Drucker, unter Verwendung der Druckereinstellungen, die im Voreinstellerprogramm Preferences festgelegt wurden. InitPrinter sendet Escape-Codes an den Druckertreiber, die bewirken, daß der Drucker die Variablen aus dem Voreinstellerprogramm erhält. Dieser Befehl kann entweder vom Arbeitstisch oder mit dem Befehlsprozessor verwendet werden.

1.3 Die Schublade Prefs

Bei der Version 1.3 befindet sich das Voreinstellerprogramm in der Schublade Prefs. Wenn Sie die Schublade öffnen, erscheint ein Fenster, das Sie zwischen fünf Piktogrammen wählen läßt:

Preferences — Öffnet das Hauptfenster des Voreinstellerprogramms
Pointer — Öffnet das Fenster für die Edierung des Zeigerpfeils
Printer — Öffnet das Fenster Change Printer
Serial — Öffnet das Fenster Change Serial
CopyPrefs — Kopiert die Systemkonfiguration nach df0:devs (dies ist nützlich für Systeme ohne automatischen Start von der Festplatte).

Sogar wenn Sie die Fenster Pointer, Printer oder Serial direkt öffnen, erscheint beim Verlassen das Hauptfenster des Voreinstellers Preferences. Sie müssen Save in diesem Hauptfenster wählen, um Änderungen, die in anderen Fenstern vorgenommen wurden, zu speichern. Nach der Wahl von Save, Use oder Cancel erfolgt die Rückkehr zum Arbeitstisch oder zum Fenster des Befehlsprozessors.

Sie können nun auch Argumente in der Befehlszeile eingeben, wenn Sie das Voreinstellerprogramm Preferences über den Befehlsprozessor aufrufen. Das Format lautet:

PREFERENCES [POINTER|PRINTER|SERIAL]

Die Option gibt das zu öffnende Fenster an.

Sowohl das Hauptfenster des Voreinstellers als auch das Fenster Change Printer wurde geändert. Die Fenster Change Serial und Edit Pointer sind identisch mit denen der Version 1.2.

1.3.1 Das Hauptfenster des Voreinstellerprogramms Preferences.

Die einzige Änderung, die im Hauptfenster des Voreinstellerprogramms vorgenommen wurde, ist die Entfernung der Option CLI ON/OFF. Auch werden, falls Ihr System mit einer batteriegepufferten Uhr ausgestattet ist, das Datum und die Uhrzeit automatisch aktualisiert, wenn das Symbol Save im Voreinsteller-Fenster gewählt wird.

1.3.2 Das Fenster Change Printer

Das Fenster Change Printer blieb unverändert, mit Ausnahme des Symbols Graphic Select, das durch zwei neue Symbole ersetzt wurde: Graphic 1 und Graphic 2. Außerdem ist der einzige auf der Workbench-Diskette vorhandene Druckertreiber der Treiber Generic. Alle anderen Druckertreiber befinden sich nun auf der Extras-Diskette. Sie müssen InstallPrinter in der Schublade Utilities verwenden, um Druckertreiber in das Verzeichnis DEVS:Printers der Workbench-Diskette zu kopieren. Welcher Treiber mit Ihrem Drucker arbeitet, entnehmen Sie bitte dem Kapitel 4, Druckertreiber.

Graphic 1

Das neue Symbol Graphic 1 entspricht dem Symbol Graphic Select der Version 1.2 und wird verwendet, um das erste der beiden Voreinstellerfenster für den Druck von Grafik aufzurufen.

Es ist eine neue Auswahl für schattierten Druck vorhanden: Gray Scale2. Diese Option für die Schattierung ist für das Drucken von Bildern vorgesehen, die mit dem Monitor A2024 erstellt wurden. (Dieser Monitor unterstützt maximal vier Graustufen.)

Graphic 2

Wird das Symbol Graphic 2 durch Anklicken gewählt, erscheint ein neues Voreinsteller-Fenster, das zusätzliche Möglichkeiten für das Drucken von Grafik unterstützt. Es stehen folgende Symbole zur Auwahl:

Smoothing [ON|OFF] — Versucht, diagonale Linien zu glätten. Diese Option ist am besten für den Einsatz mit Programmen geeignet, die Text beim Druck als Grafik betrachten (wie ProWrite, PageSetter, CityDesk, Publisher 1000, NotePad, etc.). Diese Technik verlangt etwa die doppelte Zeit für den Druck. Hinweis: Smoothing kann nicht in Verbindung mit der Einstellung F-S des Symbols Dithering gewählt werden. Wird F-S gewählt, wenn Smoothing aktiviert ist, ändert sich die Einstellung von Dithering automatisch zu Ordered (das Symbol Dithering wird später in diesem Abschnitt beschrieben). Standardeinstellung ist OFF.

Left Offset — Horizontale Verschiebung des gedruckten Bilds. Diese Option erlaubt die Einstellung eines linken Randes. Die Verschiebung kann in Vielfachen von 1/10 Inch (Zoll) eingegeben werden. Die Option Center (nachstehend beschrieben), deaktiviert die Verschiebung nach links. Standardeinstellung ist 0,0 Inch.

Center [ON|OFF] — Horizontale Zentrierung des gedruckten Bilds. Diese Option hebt den für Left Offset festgelegten Wert auf. Standardwert ist OFF.

Densitiy [1 bis 7] — Wählt die Druckdichte der Grafik. Je niedriger die Druckdichte, um so schneller der Druck (bei Druckern mit der Möglichkeit, in verschiedenen Dichten zu drucken). Die Beschreibung der Dichten, die von den einzelnen Druckern unterstützt werden, ist in den Informationen über die Druckertreiber der Version 1.3 zu ersehen (in Kapitel 4). Standardwert ist Density 1.

Color Correct R/G/B — Versucht, die Farben des Drucks besser an die der Bildschirmdarstellung anzugleichen. Diese Option versucht je nach Auswahl, alle Schattierungen von Rot (R), Grün (G) oder Blau (B) des Bildschirms mit den Druckfarben in Übereinstimmung zu bringen. Ohne Farbkorrektur (Color Correct) kann ein Farbdrucker alle 4.096. Farben des Amiga drucken. Wird jedoch die Farbkorrektur angewandt, erfolgt eine Beschränkung bis zu einem Mindestwert von 3.172 Farben (308 Schattierungen gehen für jede zur Korrektur ausgewählte Farbe verloren).

Die Auswirkung dieser Option kann sichtbar gemacht werden, indem Sie ein Bild, das gefüllte Bereiche in Rot-, Grün- und Blauschattierungen enthält, zweimal drucken – das erstemal mit Farbkorrektur, dann ohne. Vergleichen Sie nun beide Ausdrucke mit der Bildschirmdarstellung. Die Schattierungen des Ausdruck mit Farbkorrektur müssen den Farben der Bildschirmdarstellung mehr entsprechen. Der Standardwert ist "keine Korrektur" (R, G und B nicht gewählt).

Dithering [Ordered|Halftone|F-S] — Schaltet die Art der Schattierungsdarstellung ein. Bei Schattierungen werden Punkte in verschiedenen Farben so klein und eng zusammenliegend gedruckt, daß sie das Auge als nur eine Farbe erkennt. Dies ermöglicht Ihnen, Ausdrucke mit verschiedenen Schattierungen zu erzeugen, obwohl der Drucker nur vier Farben verwendet. Der Standardwert ist ORDERED. Hinweis: Ist für Shade der Wert Black and White gewählt (im Bildschirm Graphic 1) zeigt eine Änderung der Optionseinstellung von Dithering keine Auswirkung beim Ausdruck.

Ordered — Die Farbintensitäten eines Druckers werden unter Verwendung der Methode der geordneten Schattierungsdarstellung erzeugt. Farbschattierungen werden dabei durch geordnete Muster von Punkten bewirkt.

Halftone — Die Farbintensitäten eines Druckers werden unter Verwendung einer Methode der Schattierungsdarstellung erzeugt, die mit Halbtönen arbeitet. Diese Technik ist ähnlich der, die bei Zeitungen oder Comic-Heften verwendet wird. Sie arbeitet am besten mit Druckern mit hoher Druckdichte (mehr als 150 Punkte pro Inch).

F-S — Die Farbintensitäten eines Druckers werden unter Verwendung der Fehlerverteilungsmethode von Floyd-Steinberg erzeugt. Bei Wahl dieser Option dauert der Druck etwa doppelt so lange. Hinweis: Die Wahl dieser Option schaltet automatisch die Option Smoothing aus, da sie nicht gemeinsam verwendet werden können. F-S arbeitet am besten mit Druckern mit hoher Druckdichte (mehr als 150 Punkte pro Inch).

Scaling [Fraction|Integer] — Wählt die Skalierungsmethode. Standardwert ist FRACTION.

Fraction — Es wird die normale Skalierung durchgeführt.

Integer — Jeder Punkt auf dem Bildschirm wird garantiert als ganze Anzahl von Punkten auf dem Drucker dargestellt. Dies gilt für die Dimension X wie auch Y. Ist z. B. das Originalbild 320x200 Punkte groß, wird die Breite des gedruckten Bilds 320, 640, 960. etc. Punkte betragen. Die Höhe wird entsprechend mit 200, 400, 600, etc. Punkten dargestellt. Diese Option sollte gewählt werden, wenn Sie ein Bild drucken, das dünne horizontale und vertikale Linien enthält (z. B. ein Raster).

Die tatsächliche Größe des gedruckten Bilds wird die angeforderte Größe sein, vergrößert oder verkleinert zum nächsten Vielfachen der Breite und der Höhe des Bilds. Die Skalierung Integer setzt die Einstellung Aspect des Druckertreibers vollständig außer Kraft. Deshalb ist es möglich, ein leicht verzerrtes Bild (keine Korrektur des Verhältnisses der Höhe zur Breite) zu erhalten. Diese Option ist auch nützlich für das Drucken von Text als Einzelpunktgrafik (wie bei NotePad), da die Zeichen nicht durch die Skalierung Fraction verzerrt werden.

Width Limit — Begrenzt die Breite des gedruckten Bilds. Die Breite kann mit der Maßangabe 1/10 Inch (Zoll) oder Bildpunkt oder mit einem Multiplikationsfaktor festgelegt werden. Standardwert ist 0. (Siehe nachstehenden Abschnitt "Limit").

Height Limit — Begrenzt die Höhe des gedruckten Bilds. Die Höhe kann mit der Maßangabe 1/10 Inch (Zoll) oder Bildpunkt oder mit einem Multiplikationsfaktor festgelegt werden. Standardwert ist 0. (Siehe nachstehenden Abschnitt "Limit").

<- Limits [Ignore|Bounded|Absolute|Pixels|Multiply] — Bestimmt, wie die Werte für die Begrenzung der Breite und Höhe interpretiert werden. Standardwert ist IGNORE.

Ignore — Die Begrenzungen werden ignoriert (dies ist der Standardwert, der eingefügt wurde, um Kompatibilität mit Software für Versionen vor 1.3 zu erhalten). Die Größe des gedruckten Bilds weist die Größe auf, die durch das Anwendungsprogramm festgelegt ist, begrenzt durch:

Breite = (rechter Rand - linker Rand + 1) / Zeichen pro Inch
Höhe = Zeilen pro Seite / Zeilen pro Inch

Bounded — Die Größe des Bildes wird durch die Werte eingegrenzt, die bei Width Limit und Height Limit eingegeben wurden. Soll z. B. das gedruckte Bild nicht größer als 4,0 x 5,0 Inch sein (es kann jedoch kleiner sein), ist als maximale Breite 40 und als maximale Höhe 50 einzugeben und Bounded zu wählen. Diese Option wurde aufgenommen, damit die Einstellungen für Text (Ränder, Zeilen pro Zeile, etc.) nicht bei jedem Druck einer Grafik geändert werden müssen.

Absolute — Die Begrenzungen der Breite und der Höhe werden als Absolutwerte aufgefaßt. Die Größe des Bildes ist nicht mehr nur nach oben begrenzt, sondern es wird die absolute Größe angegeben. Wenn z. B. ein Bild genau 4,0 x 5,0 Inch groß sein soll, ist als maximale Breite 40 und als maximale Höhe 50 einzugeben und Absolute zu wählen. Dadurch wird die Einstellung Aspect des Druckertreibers vollständig außer Kraft gesetzt. Es ist deshalb möglich, daß Sie ein sehr verzerrtes Bild erhalten (keine Korrektur des Verhältnisses von Höhe und Breite).

Manchmal kann es erforderlich sein, den Ausdruck mit einer speziellen Breite oder Höhe festzugelegen, wobei das Verhältnis der Höhe zur Breite nicht verändert, das Bild also nicht verzerrt werden soll. In diesem Fall setzen Sie die Höhen- oder die Breitenbegrenzung auf den gewünschten Wert und den anderen Wert zu Null. Ist eine der Dimensionen mit Null angegeben, wird diese Dimension entsprechend der festgelegten Dimension korrigiert. Wurde z. B. die Breitenbegrenzung mit 40 angegeben und die Höhenbegrenzung mit 0, ist das gedruckte Bild 4,0 Inch breit und so hoch wie erforderlich, um das korrekte Dimensionsverhältnis zu erreichen.

Werden beide Dimensionen mit Null angegeben, wird das Bild so breit gedruckt, wie es die maximale Punktanzahl des Druckers zuläßt und so hoch, daß das Verhältnis der Dimensionen korrekt ist.

Pixels — Die Werte der Breiten- und Höhenbegrenzung werden als Anzahl der Punkte verstanden, anstatt als Mehrfaches von 1/10 Inch. Es gelten dieselben Regeln wie für die Option Absolute.

Multiply — Die Werte der Breiten- und Höhenbegrenzung werden als Multiplikator für die Breite und Höhe des Ausgangsbilds verstanden. Wird z. B. Width Limit mit 4 und Height Limit mit 2 angegeben, wird das gedruckte Bild die doppelte Breite des Ausgangsbilds (in Bildpunkten) und die vierfache Höhe aufweisen. Deshalb wird, wenn das Ausgangsbild 320 x 200 Bildpunkte groß ist, das gedruckte Bild 640 (320 x 2) Punkte breit und 800 (200 x 4) Punkte hoch sein. Es gelten dieselben Regeln wie für die Option Absolute (Höhenbegrenzung = 0 und Breitenbegrenzung = 0). Außerdem sind hierbei die Vorteile, die die Skalierung Integer bietet, wirksam.

1.3.3 Das Piktogramm CopyPrefs

Das Doppelklicken auf dieses Piktogramm ruft ein Fenster mit der Meldung "Copying devs:system-configuration to DF0:DEVS" auf den Bildschirm. Dies ist für Benutzer gedacht, bei denen DEVS: einem anderen als dem Verzeichnis devs auf der Neustart-Diskette zugewiesen ist. Zum Beispiel Benutzer, deren Festplatte nicht automatisch für den Neustart verwendet wird und die diesen mit einer Diskette durchführen müssen (d. h. Benutzer von A2090). Mit Preferences geänderte Voreinstellungen werden dann auf der Festplatte statt auf der Workbench-Diskette gespeichert, wo sie eigentlich gebraucht werden. Wenn Sie die Voreinstellungen auf der Festplatte verändert haben und die Workbench-Diskette für den Neustart aktualisieren wollen, verwenden Sie CopyPrefs. CopyPrefs kopiert die Datei System-Configuration von der Festplatte auf die Workbench-Diskette. Beim Start von CopyPrefs muß sich die korrekte Diskette im Laufwerk df0: befinden.

1.4 Die Hilfsprogrammschublade (Utilities)

Der Umfang der Hilfsprogrammschublade auf der Workbench-Diskette der Version 1.3 hat sich seit der Freigabe der Version 1.2 mehr als verdreifacht. Die Hilfsprogramme Say und GraphicDump wurden in der Schublade Utilities untergebracht, zusammen mit den vertrauten Hilfsprogrammen NotePad, Calculator

und Clock. Außerdem enthält sie nun fünf neue Programme — ClockPtr, CMD, InstallPrinter, MORE und PrintFiles.

- ClockPtr — gestaltet den Zeiger zu einer Digitaluhr um
- CMD — leitet die serielle oder parallele Ausgabe in eine Datei um
- InstallPrinter — kopiert Druckertreiber von der Extras-Diskette zur Workbench-Diskette
- MORE — erlaubt die Anzeige von ASCII-Textdateien in einem Workbench-Fenster
- PrintFiles — erlaubt das Drucken mehrerer Dateien in Folge

1.4.1 ClockPtr

Das Programm ClockPtr ändert Ihren Workbench-Zeigerpfeil zu einer Digitaluhr, wann immer der Hintergrundbildschirm des Arbeitstischs aktiv ist. Soll ClockPtr vom Arbeitstisch aus gestartet werden, doppelklicken Sie auf das Piktogramm. Bewegen Sie den Zeiger aus dem Utilities-Fenster, drücken Sie die Maustaste und der Zeiger wird zur Uhr. ClockPtr zeigt die Stunde und die Minute an.
Sollen Minuten und Sekunden angezeigt werden, bewegen Sie den Zeiger zur linken oberen Ecke des Arbeitstischs, und ein Pausenzeitmesser wird angezeigt. Für die Anzeige des Datums bewegen Sie den Zeiger zur linken Seite des Arbeitstischs.

ClockPtr kann durch Doppelklicken auf das Piktogramm geschlossen werden oder durch Drücken von Control-C, sofern es über den Befehlsprozessor gestartet wurde. Wird ClockPtr im Befehlsprozessor durch den AmigaDOS-Befehl RUN gestartet, muß der Befehl BREAK zum Schließen verwendet werden.

Für das Einstellen der Uhrzeit verwenden Sie den Befehl DATE oder SETCLOCK (siehe Kapitel 2).

1.4.2 CMD

CMD_Write ist eine interne Amiga-Funktion, die das Verhalten des Computers steuert. Wenn Sie ein Softwareprogramm oder einen CLI-Befehl verwenden, um Informationen an Ihren Drucker zu senden, wird der Amiga durch eine Funktion CMD_Write über Ihre Entscheidung informiert. Durch das Hilfsprogramm CMD kann die serielle oder parallele Ausgabe an ein Gerät, die über CMD_Write erfolgt, in eine gewünschte Datei umgeleitet werden. Dadurch wird es möglich, die Druckausgabe in eine Datei aufzunehmen.

Soll das Hilfsprogramm CMD über den Befehlsprozessor gestartet werden, gilt folgendes Befehlsformat:

CMD <Gerätename> <Dateiname> [OPT s|m|n]

Für <Gerätename> wird entweder "serial" oder "parallel" angegeben und für <Dateiname> der Name der Datei, zu der die umgeleitete Ausgabe gesendet werden soll. Beachten Sie, daß der Befehl CMD die Ausdrücke "PAR:" oder "SER:" nicht als Gerätenamen akzeptiert.

Die CMD-Optionen lauten wie folgt:

s Überspringt alle einleitenden Steuerdaten des Schreibvorgangs (normalerweise ein Rücksetzbefehl, wenn der Ausdruck eines Bildschirms umgeleitet wird)*.

m Legt diese Einstellung für mehrere Ausgaben fest (MULTIPLE), bis ein Befehl BREAK gegeben oder Control-C gedrückt wird.

n Aktiviert NOTIFY (nützliche Bildschirmmeldungen beim Ablauf)

* Beim Ausdruck eines Bildschirms wird durch das erste CMD-Write normalerweise eine Druckerrücksetzung vorgenommen. Der Druckertreiber verzögert dann den Ablauf so lange, bis die Rücksetzung abgeschlossen ist. Dadurch wird der Verlust nachfolgender Ausgabedaten verhindert. Wenn das Hilfsprogramm CMD verwendet wird, um die Ausgabe in eine Datei umzuleiten, findet diese Verzögerung nicht statt. Gibt Ihr Druckertreiber am Beginn eines Ausdrucks einen Rücksetzbefehl aus, müssen Sie die Option s verwenden, damit das erste CMD_Write nicht in die Datei geschrieben wird.

Sie können CMD auch über den Arbeitstisch aufrufen. Sie geben dann den Gerätenamen, Dateinamen und die Optionen über die TOOL TYPES des Piktogramms ein. Wählen Sie das CMD-Piktogramm und wählen Sie dann INFO aus dem Workbench-Menü. CMD erkennt die folgenden Parameter im Feld TOOL TYPES des CMD-Piktogramms:

DEVICE= Setzen Sie entweder parallel oder seriell, Standardwert ist parallel.

FILE= Der Name der Zieldatei, Standardwert ist ram:cmd_file.

SKIP= Setzen Sie TRUE, um die kurzen Schreibvorgänge am Anfang zu überspringen. Standardwert ist FALSE — einleitende Schreibvorgänge werden nicht übersprungen.

MULTIPLE= Setzen Sie TRUE, um den Vorgang für mehrere Ausgaben zu aktivieren. Standardwert ist FALSE — nur eine Ausgabe.

NOTIFY= Setzen Sie TRUE, um Meldungen über die Abarbeitung des Programms zu erhalten. Standardwert ist FALSE — es werden keine Meldungen gesendet.

Das Doppelklicken auf das Piktogramm CMD bewirkt, daß das Hilfsprogramm alle im Feld TOOL TYPES festgelegten Werte für die Umleitung übernimmt.

1.4.3 InstallPrinter

InstallPrinter ist ein Programm, das Ihnen das Kopieren eines Druckertreibers von der Extras-Diskette zum Verzeichnis devs:printers auf der Workbench-Diskette erlaubt, ohne dafür einen AmigaDOS-Befehl zu verwenden oder eine SHELL oder einen CLI zu öffnen. Doppelklicken Sie auf das Piktogramm InstallPrinter. Es erscheint ein Fenster, in dem alle Druckertreiber aufgelistet werden, die sich auf der Extras-Diskette befinden. Geben Sie den Namen des gewünschten Treibers in der Zeile der Eingabeaufforderung ein und drücken Sie RETURN. Die Datei des Druckertreibers wird automatisch zum Verzeichnis devs:printers auf der Workbench-Diskette kopiert, und der Name des Treibers erscheint im Fenster Change Printer des Voreinstellerprogramms Preferences. Sie können für das Kopieren mehrerer Treiber auch Jokerzeichen verwenden. Soll InstallPrinter verlassen werden, ohne einen Treiber zu kopieren, drücken Sie die Taste RETURN.

1.4.4 MORE

MORE ist ein Hilfsprogramm für die Anzeige von ASCII-Textdateien. Es gibt zwei Möglichkeiten, MORE vom Arbeitstisch aus zu starten. Ist die Textdatei mit einem Piktogramm versehen, wählen Sie das Piktogramm MORE und doppelklicken dann auf das Piktogramm der Textdatei, während Sie die Taste Shift gedrückt halten. Ist kein Piktogramm für die Textdatei vorhanden, Doppelklicken Sie auf das MORE-Piktogramm. Es erscheint eine Eingabeaufforderung, die Sie nach dem Namen der Textdatei fragt. Geben Sie unbedingt den gesamten Pfad an.

Soll MORE über den Befehlsprozessor gestartet werden, lautet das Befehlsformat wie folgt:

MORE <Dateiname>

Bei der Angabe der anzuzeigenden Datei muß sichergestellt sein, daß der vollständige Pfad inklusive des Datenträgernamens (oder der Laufwerksnummer), des Verzeichnisses und des Dateinamens eingegeben wird. Wenn Sie keine Datei angeben, wird MORE eine kurze Erklärung anzeigen und eine Eingabeaufforderung, die Sie nach dem Dateinamen fragt.

Während des Ablaufs von MORE erscheint eine Meldung wie z. B. "--- More (48%) ---" am unteren Rand des Fensters. Sie zeigt, wieviel Prozent der Datei bis jetzt angezeigt wurden. Für das Blättern durch die Anzeige können die folgenden Tastenkombination verwendet werden:

<LEERTASTE>	Zeigt die nächste Seite
<BACKSPACE>	Zeigt die vorhergehende Seite*
<RETURN>	Zeigt die nächste Zeile
<	Zeigt die erste Seite*
>	Zeigt die letzte Seite*
%n	Zeigt den Inhalt der Datei bei ca. n %*
CTRL + L	Aktualisiert die Anzeige im Fenster
/<Zeichenfolge>	MORE sucht nach der Zeichenfolge, die auf den "/" folgend angegeben wurde. Groß- und Kleinschreibung wird berücksichtigt.
.<Zeichenfolge>	MORE sucht nach der Zeichenfolge, die angegeben wurde. Groß- und Kleinschreibung wird nicht berücksichtigt.
n	Das nächste Auftreten der Zeichenfolge wird gesucht.
h	Hilfe (zeigt eine Liste ähnlich dieser an)
q	Beenden (oder Control-C)
E	Bearbeitung unter Verwendung des Editors, der in ENV: EDITOR festgelegt ist.

* MORE akzeptiert auch Eingaben über PIPE. Da bei der Standardeingabe über den PIPE-Handler die Länge unbekannt ist, sind die mit einem Stern gekennzeichneten Befehle deaktiviert, wenn die MORE-Eingabe über PIPE erfolgt. (Siehe Kapitel 3, Andere Workbench-Verzeichnisse).

Das Suchen unter Berücksichtigung der Groß- und Kleinschreibung bedeutet, daß MORE nach einer Zeichenfolge sucht, die genau der entspricht, die eingegeben wurde. Enthält die eingegebene Zeichenfolge Großbuchstaben, sucht MORE nur nach einer identischen Zeichenfolge, die ebenfalls diese Großbuchstaben beinhaltet. Beim Suchen ohne Berücksichtigung der Groß-/Kleinschreibung spielt es keine Rolle, ob die Groß- und Kleinschreibung beim gesuchten und beim gefundenen Begriff übereinstimmen.

Wenn Sie die letzte Seite der Anzeige erreichen, erscheint am unteren Bildschirmrand die Nachricht "--- Less ---".

Wird MORE über den CLI oder die SHELL aufgerufen, kann ein Editor verwendet werden, um die angezeigte Datei zu verändern (Shift-E drücken), sofern die Umgebungsvariable für EDITOR definiert ist. In der EDITOR-Variablen muß der vollständige Pfad zum Editor angegeben sein, d. h. Extras:Tool/MEMACS oder C:ED. (Weitere Informationen über die Verwendung der Umgebungsvariablen sind im Abschnitt über den RAM-Handler in diesem Kapitel und in den Abschnitten über die Befehl GETENV und SETENV in Kapitel 2 zu finden.)

HINWEIS: Wenn Sie bei Verwendung von MORE versehentlich die Taste Escape drücken, erfolgt eine Unterbrechung. Für die Fortsetzung drücken Sie entweder die Taste "q" oder die Taste "h".

1.4.5 PrintFiles

Mit PrintFiles können Sie Dateien zum Drucker kopieren. PrintFiles akzeptiert auch mehrere Dateinamen, so daß eine Folge von zu druckenden Dateien festgelegt werden kann. Kann PrintFiles eine dieser Dateien nicht finden oder öffnen, wird diese Datei übergangen und mit der nächsten fortgefahren.

Für den Aufruf von PrintFiles über den Befehlsprozessor gilt folgendes Befehlsformat:

PRINTFILES [-f] <Dateiname> [[-f] <Dateiname>] [[-f] <Dateiname>] . . .
Das Kennzeichen -f bewirkt, daß ein Seitenvorschub zwischen den aufeinanderfolgenden Dateien und nach dem Ende der letzten Datei ausgeführt wird. Soll der Seitenvorschub aktiviert werden, wenn PrintFiles über den Arbeitstisch aufgerufen wird, müssen Sie den Befehl INFO aus dem Workbench-Menü verwenden, um das INFO-Fenster von PrintFiles zu öffnen. Fügen Sie dann "FLAGS=FORMFEED" den TOOL TYPES des Piktogramms hinzu.

Verwendung von PrintFiles über den Arbeitstisch:

1) Wählen Sie das Piktogramm der ersten zu druckenden Datei.
2) Halten Sie die Taste Shift gedrückt, und wählen Sie die Piktogramme aller zusätzlich zu druckenden Dateien.
3) Halten Sie die Taste Shift gedrückt, und doppelklicken Sie auf das Piktogramm von PrintFiles.

1.4.6 GraphicDump

GraphicDump sendet den Bildschirm im Vordergrund als Grafik zum Drucker. Dies erfolgt ca. zehn Sekunden nach dem Doppelklicken auf das Piktogramm. (Die Verzögerung von 10 Sekunden erlaubt Ihnen, die Bildschirme so anzuordnen, daß sich der zu druckende Bildschirm im Vordergrund befindet.) Es bestehen einige Unterschiede zum Programm GraphicDump der Version 1.2: Die Größe der erzeugten Druckausgabe kann nun festgelegt werden. Soll GraphicDump über den Befehlsprozessor aufgerufen werden, lautet das Befehlsformat wie folgt:

GraphicDump [TINY|SMALL|MEDIUM|LARGE|x-Punkte:y-Punkte]

Die gewünschte Größe des Ausdrucks kann durch eine der fünf Optionen festgelegt werden:

TINY	Die Breite des Ausdrucks umfaßt etwa ein 1/4 der maximalen Druckbreite des Druckers. Die Höhe entspricht dem Verhältnis von Höhe und Breite, das der Bildschirm bei der Anzeige aufweist.
SMALL	Die Breite des Ausdrucks umfaßt etwa die Hälfte der maximalen Druckbreite des Druckers. Die Höhe entspricht dem Verhältnis von Höhe und Breite, das der Bildschirm bei der Anzeige aufweist.
MEDIUM	Die Breite des Ausdrucks umfaßt etwa 3/4 der maximalen Druckbreite des Druckers. Die Höhe entspricht dem Verhältnis von Höhe und Breite, das der Bildschirm bei der Anzeige aufweist.

LARGE	Die Breite des Ausdrucks umfaßt die maximale Druckbreite des Druckers. Die Höhe entspricht dem Verhältnis von Höhe und Breite, das der Bildschirm bei der Anzeige aufweist. (Wird GraphicDump über den Arbeitstisch aufgerufen, ist LARGE der Standardwert für den Ausdruck).
x-Punkte:y-Punkte	Geben Sie durch x-Punkte die absolute Breite in Punkten und durch y-Punkte die absolute Höhe in Punkten an. Dies erlaubt dem Benutzer, jegliche Dimension innerhalb der Grenzen des Druckers anzugeben.

Damit GraphicDump die Optionen TINY, SMALL, MEDIUM und LARGE erkennen kann, muß als Einstellung für LIMITS im Voreinstellerprogramm Preferences die Option IGNORE (Standardwert) gewählt sein. Ist für LIMITS eine andere Option angegeben, wird die Größe des Ausdrucks durch diese Option bestimmt. Bei den folgenden Beispielen wird vorausgesetzt, daß für LIMITS die Option IGNORE eingestellt ist.

1) Soll der Ausdruck des Fensters im Vordergrund etwa die Hälfte der maximalen Druckbreite des Druckers umfassen, ist der folgende Befehl einzugeben:

```
1>GRAPHICDUMP SMALL
```

2) Sollen die exakten Abmessungen des Ausdrucks mit z. B. 640 Punkten Breite und 200 Punkten Höhe festgelegt werden, ist der folgende Befehl einzugeben:

```
1>GRAPHICDUMP 640:200
```

1.4.7 Die Uhr (Clock)

Sobald Sie die Uhr öffnen, erscheint eine Analoguhr mit den Abmessungen von 170 x 90 Bildpunkten am Bildschirm. Wird die Uhr als Analoguhr angezeigt, kann sie unter Verwendung des Größen-Symbols vergrößert oder verkleinert werden. Die Größe der Uhr kann nur verändert werden, solange sie als Analoguhr angezeigt wird.

Wenn Sie eine Digitaluhr bevorzugen, stehen Ihnen nun zwei Möglichkeiten zur Auswahl: Digital 1 oder Digital 2. Digital 1 zeigt die Uhr in einem rechteckigen

Fenster mit zwei Zeilen — einer für die Zeit und einer für das Datum. Digital 2 erzeugt ein kleineres Fenster, das in der Titelleiste erscheint und so keine anderen Anzeigen am Bildschirm überlagert. Ist die Option Datum aktiviert, werden in diesem Fenster das Datum und die Uhrzeit abwechselnd angezeigt.

Der Wecker (Alarm) arbeitet genau wie bisher, mit einem kleinen Unterschied. Wenn nun das Alarmfenster erscheint, bleibt die Uhr in Funktion. Bei der Version 1.2 blieb die Anzeige der Uhr stehen, wenn das Kommunikationsfenster für den Alarm angezeigt wurde. Beachten Sie, daß Sie die Uhr nicht schließen können, wenn ein Alarm eingestellt ist.

Wenn Sie die Uhr über den Befehlsprozessor starten, können Sie Argumente in der Befehlszeile eingeben, die die Art der gewünschten Uhr angeben. Das Befehlsformat lautet wie folgt:

```
CLOCK  [ANALOG|DIGITAL1|DIGITAL2] [=<x>,<y>
       [,<Breite>, <Höhe>]] [12HOUR|24HOUR]
       [SECONDS] [DATE]
```

Die Optionen bedeuten folgendes:

ANALOG, DIGITAL1, DIGITAL2 — Beziehen sich auf die Art der Anzeige wie oben beschrieben. Standardwert ist ANALOG.

=<x>,<y> — Erlaubt Ihnen festzulegen, wo am Bildschirm die Uhr angezeigt werden soll. Die für <x> angegebene Zahl gibt den horizontalen Abstand zum linken Bildschirmrand an, während der Abstand zum oberen Bildschirmrand durch <y> festgelegt wird. Die Optionen <Höhe> und <Breite> beziehen sich auf die Größe der Analoguhr. Sie wird in Bildpunkten angegeben. Diese Optionen werden ignoriert, wenn sie für eine Digitaluhr angegeben werden. Stellen Sie beim Eingeben der Befehlszeile sicher, daß sich zwischen dem Wort ANALOG und dem Gleichheitszeichen kein Leerschritt befindet. Standardwert ist =10,15.

12HOUR oder 24HOUR — Sie können zwischen einer Uhr mit 12-Stunden-Anzeige (mit Angabe von AM und PM) oder 24-Stunden-Anzeige wählen. Standardwert ist 12 Stunden.

SECONDS — Zeigt die Sekunden. Standardwert für diese Option ist OFF.

DATE — Zeigt das Datum. Standardwert für diese Option ist OFF.

Zum Beispiel wird nach:

```
1> Run Clock ANALOG=10,20,600,160 12HOUR DATE
```

eine Analoguhr angezeigt, die folgende Merkmale aufweist: 10 Punkte vom linken und 20 Punkte vom oberen Bildschirmrand entfernt, 600 Punkte breit und 160 Punkte hoch, für Anzeige von 12 Stunden (mit Kennzeichnung AM und PM) und Datumsanzeige.

1.5 Die beiden RAM-Disks

Auf dem neuen Arbeitstisch stehen zwei RAM-Disks zur Verfügung. Der standardgemäße RAM-Handler (RAM:), für den automatisch ein Piktogramm auf dem Arbeitstisch erscheint, und die neue resetfeste RAM-Disk (RAD:).

1.5.1 RAM-Handler

Der RAM-Handler, der sich im Verzeichnis L: der Workbench-Diskette befindet, ist die standardgemäße RAM:-Disk, die beim Amiga bereits immer vorhanden war. Das Piktogramm dieses Steuerprogramms erscheint in der rechten oberen Ecke des Arbeitstischs als RAM DISK. Der Amiga erkennt diese "Diskette" unter dem Namen RAM:, ebenso wie er ein internes Laufwerk unter dem Namen df0: erkennt. RAM: ändert ihre Größe je nach Inhalt. Deshalb ist sie immer zu 100 % voll. Werden weitere Dateien dort gespeichert, wird sie erweitert bzw. beim Löschen von Dateien verkleinert.

Alle bekannten Fehler des RAM-Handlers wurden bei der Version 1.3 beseitigt. Die deutlichsten Unterschiede sind:

- Der RAM-Handler wird nun "RAM DISK" genannt.
- Er kann unter Verwendung des Befehls RENAME aus dem Workbench-Menü umbenannt werden.

Während des Starts des Arbeitstisches werden verschiedene Verzeichnisse in RAM: erstellt, dann die logischen Gerätenamen T:, ENV: und CLIPS: zugewiesen. Das Verzeichnis T: von RAM: wird von Befehlsdateien verwendet, während CLIPS: ein neues, alternatives Verzeichnis für die Speicherung von Zwischenspeicherinhalten ist. Sie können diese Verzeichnisse in RAM: belassen oder, falls gewünscht, an anderer Stelle zuweisen.

Im Verzeichnis ENV: werden die Umgebungsvariablen gespeichert, die durch den Befehl SETENV erstellt werden. Sie verwenden SETENV, um einer Variablen einen "Wert" zuzuweisen. Das ist nicht unbedingt eine mathematische oder numerische Variable, sondern es kann auch eine veränderliche Zeichenfolge sein. Wenn Sie z. B. die Option Editor des Programms MORE verwenden wollen, und als Edierprogramm MEMACS eingesetzt werden soll, ist folgender Befehl einzugeben:

```
SETENV EDITOR "Extras 1.3:Tools/MEMACS"
```

In diesem Fall stellen der Name und der Pfad zum Edierpogramm den Inhalt der Variablen dar. Wenn Sie das Verzeichnis ENV: nach Verwendung dieses Befehls SETENV anzeigen lassen, finden Sie dort eine Datei mit dem Namen EDITOR. Momentan werden Umgebungsvariablen durch das Hilfsprogramm MORE und den Befehl IF verwendet. Künftig werden auch andere Befehle damit arbeiten.

Verwenden Sie den Befehl GETENV, wenn eine Variable unter Verwendung von SETENV erstellt wurde, und Sie später die Zeichenfolge sehen wollen, die dieser Variablen zugewiesen wurde. Geben Sie GETENV ein, gefolgt vom Namen der gewünschten Variablen, und das System antwortet mit dem Wert oder der Zeichenfolge, die diesem Namen zugewiesen wurde.

1.5.2 Die resetfeste RAM-Disk

Eine resetfeste RAM-Disk erlaubt Ihnen, nach dem Neustart Ihres Systems auf ihren Inhalt zuzugreifen. Sogar, wenn der Neustart mit einer anderen als der Workbench-Diskette erfolgte (z. B. einem Textverarbeitungsprogramm), sind die resetfeste RAM-Disk und ihr Inhalt noch vorhanden. Der Inhalt geht nicht verloren, bis der Amiga ausgeschaltet oder der Befehl REMRAD verwendet wird.

Ebenso wie ein Festplattenlaufwerk oder ein anderes Erweiterungsgerät muß die resetfeste RAM-Disk durch den Befehl MOUNT installiert werden. Es befindet sich ein einzelner Eintrag (mit dem Namen RAD:) in der Datei MountList im Verzeichnis DEVS: der Workbench-Diskette. Nachstehend ist ein Beispiel einer Befehlsprozessor-Sitzung aufgeführt, welche die oben beschriebenen Abläufe durchführt:

```
1> TYPE devs/MountList
```

Nur um sicherzustellen, daß sich der Eintrag RAD: in der Datei befindet. Beim Standardeintrag ist Anzahl der Zylinder mit 21 festgelegt. Zum Vergleich: Eine

standardgemäße Amiga-Diskette verfügt über 79 Zylinder. Soll die Kapazität der resetfesten RAM-Disk erhöht werden, ist ein höherer Wert für die Anzahl der Zylinder einzugeben.

```
1> MOUNT rad:
```

Teilt dem Amiga mit, daß die resetfeste RAM-Disk verwendet wird. Für die Prüfung, ob die RAD: korrekt angemeldet wurde, verwenden Sie den Befehl INFO.

```
1> RELABEL DRIVE rad: NAME MyRad
```

Sie können den Namen des Geräts (Schlüsselwort DRIVE) beliebig wählen, mit Ausnahme der Bezeichnung RAM. Es könnte sonst mit der anderen RAM-Disk des Arbeitstisches verwechselt werden.

```
1> DISKCHANGE Rad:
```

Informiert den Arbeitstisch über die mit RELABEL vorgenommene Namensänderung.

Wurde Ihr Computer nicht mit der Kickstart-Version 1.3 aktualisiert, entweder durch Verwendung der neuen Kickstart-Diskette 1.3 (beim Amiga 1000) oder durch Einbau des neuen Kickstart-ROM (beim Amiga 500 oder Amiga 2000), muß der Befehl MOUNT RAD: nach jedem Neustart angegeben werden.

Soll RAD: entfernt werden, ohne das gesamte System auszuschalten, können Sie den Befehl REMRAD verwenden. Geben Sie bei der Eingabeaufforderung des Befehlsprozessors REMRAD ein und die resetfeste RAM-Disk (RAD:) wird auf ein Minimum verkleinert. Nach dem Neustart des Systems wird RAD: dann nicht erscheinen.

Bei Verwendung der Kickstart-Version 1.3 ist es möglich, den Neustart direkt von RAD: auszuführen, ohne die Workbench-Diskette oder eine andere für den Neustart geeignete Diskette in das Laufwerk einzulegen. (Siehe Kapitel 5, Ändern der Startsequenzen.)

2. AmigaDOS und das Verzeichnis C der Workbench-Diskette

In diesem Kapitel werden alle Änderungen oder Zusätze angegeben, die bezüglich der Befehle im Verzeichnis C der Workbench-Diskette vorgenommen wurden. Es werden auch einige neue Befehle in diesem Kapitel beschrieben. Weitere Informationen über AmigaDOS entnehmen Sie bitte dem entsprechenden Kapitel des Benutzer-Handbuchs oder dem vom Verlag Markt & Technik herausgegebenen *AmigaDOS-Handbuch*, das bei vielen Computer- und Buchhändlern erhältlich ist.
Im Verzeichnis C der Workbench-Diskette sind 64 Befehle vorhanden:

ADDBUFFERS	EDIT	JOIN	RESIDENT
ASK	ELSE	LAB	RUN
ASSIGN	ENDCLI	LIST	SEARCH
AVAIL	ENDIF	LOADWB	SETCLOCK
BINDDRIVERS	ENDSKIP	LOCK	SETDATE
BREAK	EVAL	MAKEDIR	SETENV
CD	EXECUTE	MOUNT	SETPATCH
CHANGETASKPRI	FAILAT	NEWCLI	SKIP
COPY	FAULT	NEWSHELL	SORT
DATE	FF	PATH	STACK
DELETE	FILENOTE	PROMPT	STATUS
DIR	GETENV	PROTECT	TYPE
DISKCHANGE	ICONX	QUIT	VERSION
DISKDOCTOR	IF	RELABEL	WAIT
ECHO	INFO	REMRAD	WHICH
ED	INSTALL	RENAME	WHY

Viele dieser Befehle werden nur von Programmierern und fortgeschrittenen Amiga-Benutzern verwendet. Es finden jedoch auch viele "Durchschnittsbenutzer" AmigaDOS hilfreich, um die auf den Disketten enthaltenen Informationen zu organisieren. Aus diesem Grund werden hierin alle neuen oder geänderten Befehle beschrieben, so daß alle Benutzer auf diese Informationen zugreifen können.

2.1 Unveränderte Befehle

Diese Befehle wurden in der Version 1.3 nicht geändert und sind im AmigaDOS-Handbuch vollständig beschrieben.

BREAK	Setzt Unterbrechungskennungen im angegebenen Prozeß.
CD	Für das Setzen, Ändern oder Anzeigen des aktuellen Verzeichnisses
ED	Für das Edieren von Textdateien (ein Bildschirmeditor)
EDIT	Für das Edieren von Textdateien, wobei die Quellendatei sequentiell bearbeitet wird (ein Zeileneditor)
ELSE	Für die Angabe von Alternativen bei der bedingten Verarbeitung in einer Befehlsdatei
ENDIF	Der Abschluß eines IF-Blocks in einer Befehlsdatei
FAILAT	Für das Setzen von Fehlerbedingungen in einer Befehlsdatei
FAULT	Für den Druck der angegebenen Fehlermeldung(en)
LAB	Für die Angabe eines Markennamens in einer Befehlsdatei.
MAKEDIR	Für das Erstellen eines neuen Verzeichnisses
QUIT	Für das Verlassen einer Befehlsdatei mit einem angegebenen Fehlercode
RELABEL	Für das Ändern eines Diskettennamens
RENAME	Für das Ändern eines Datei- oder eines Verzeichnisnamens
SORT	Für das Sortieren kleiner Dateien
STACK	Für das Anzeigen oder Setzen der Stack-Speichergröße für den aktuellen Befehlsprozessor
WAIT	Bewirkt, daß AmigaDOS eine angegebene Zeit in Wartezustand versetzt wird.
WHY	Für den Druck einer Fehlermeldung, die erklärt, warum der vorausgegangene Befehl einen Fehler erzeugte.

2.2 Neue oder geänderte Befehle

Viele AmigaDOS-Befehle der Version 1.2 sind in Version 1.3 geändert, andere waren in der verfügbaren Dokumentation nicht vollständig beschrieben. Diese Befehle werden unter Verwendung der AmigaDOS-Standarddarstellung nachstehend aufgelistet: Format, Schablone, Zweck, Pfad und Beschreibung. Einige der Schreibweisen, mit denen Sie vertraut sein müssen, sind:

BEFEHL — AmigaDOS-Befehle werden in Großbuchstaben geschrieben, um sie vom restlichen Text abzuheben.

< > — Spitze Klammern schließen Argumente ein, die angegeben werden müssen. <Dateiname> bedeutet z. B., daß ein spezifischer Dateiname zusammen mit dem AmigaDOS-Befehl eingegeben werden muß.

[ ] — Eckige Klammern schließen Optionen ein. Optionen werden durch das AmigaDOS akzeptiert, sind aber nicht zwingend erforderlich.

{ } — Geschweifte Klammern bedeuten, daß die eingeschlossenen Befehlsteile beliebig oft wiederholt werden können, die Eingabe jedoch nicht zwingend ist. {<Argument>} bedeutet z. B., daß beliebig viele Argumente eingegeben werden können, sie jedoch nicht unbedingt erforderlich sind.

. . . — Drei Punkte zeigen, daß eine Folge fortgesetzt werden kann.
* — Gleiche Bedeutung wie drei Punkte...

| — Ein Senkrechtstrich wird verwendet, um Optionen in einer Liste zu trennen, von denen jeweils eine benutzt werden kann. [OPT R|S|RS] bedeutet z. B., daß die Option R, die Option S oder beide (RS) gewählt werden können.

<Argument>/A — Das **Argument** muß angegeben werden, damit der Befehl arbeitet.

<Option>/K — Das **Schlüsselwort** (K wie Key) der Option muß eingegeben werden, auch wenn weitere Parameter angegeben sind. Eine der Optionsgruppen des Befehls DIR ist [OPT A|I|AI|D]. Die Schablone lautet "OPT/K". Das bedeutet, daß das Wort OPT zusammen mit der gewählten Option eingegeben werden muß, also entweder OPT A, OPT I, OPT AI oder OPT D. Es darf nicht nur A, I, AI oder D alleine geschrieben werden.

<Option>/S — Die Option arbeitet als **Schalter**. Der Name der Option muß eingegeben werden, um sie zu wählen. Der Befehl DIR verfügt z. B. auch über eine Option [ALL]. Das Wort ALL muß geschrieben werden, wenn Sie diese Option benutzen wollen.

Einige der Befehlserklärungen beziehen sich auf Fehlercodes von Befehlen. Ein Fehlercode wird gesetzt, wenn ein Befehl nicht korrekt durchgeführt werden kann. Der Fehlercode 5 repräsentiert z. B. eine Warnung und bedeutet, daß, obwohl ein Fehler aufgetreten ist, dieser nicht schwerwiegend genug war, um die Verarbeitung abzubrechen. Nachfolgende Befehle werden ausgeführt. Weitere Erklärungen zu diesem Thema sind im Abschnitt über den Befehl FAILAT im AmigaDOS-Handbuch aufgeführt.

ADDBUFFERS

Format: ADDBUFFERS <Laufwerk>: <nn>

Schablone: ADDBUFFERS "LAUFWERK/A,PUFFER/A"

Zweck: Befehl an das System, weitere Cache-Puffer anzulegen.

Pfad: C:ADDBUFFERS

Bedeutung:

ADDBUFFERS erweitert die Liste der Sektor-Cache-Puffer für <Laufwerk> um <nn> Puffer. Bei Verwendung des Standard-Dateisystems können zusätzliche Puffer die Zugriffszeit auf die entsprechende Diskette erheblich verkürzen. Jeder zusätzliche Puffer belegt jedoch ca. 500 Byte im Hauptspeicher. Im allgemeinen sollten nicht mehr als 25 bis 30 Puffer hinzugefügt werden. Bei höheren Werten entsteht kein weiterer wesentlicher Zeitvorteil, es wird jedoch wertvoller Speicher verschwendet.

Wird mit FastFileSystem auf einer Festplatte gearbeitet, beschleunigt das Hinzufügen zusätzlicher Puffer immer den Plattenzugriff. (Weitere Informationen zu FastFileSystem sind im Anhang A zu finden.) Die empfohlene Anzahl der zusätzlichen Puffer hängt davon ab, wie viel zusätzlicher Speicher verfügbar ist.

Beispiel:

```
1> ADDBUFFERS df1: 25
```

Fügt zu den Sektor-Cache-Puffern für das Diskettenlaufwerk df1: 25 Puffer hinzu.

ASK

Format: ASK <Eingabeaufforderung>

Schablone: ASK "EINGABEAUFFORDERUNG/A"

Zweck: Ermöglichung von Benutzereingaben bei der Ausführung einer *Befehlsdatei.*

Pfad: C:ASK

Bedeutung:

ASK schreibt die <Eingabeaufforderung> in den aktuellen Ausgabestrom und wartet dann auf eine Tastatureingabe durch den Benutzer. Gültige Anworten sind Y (Ja), N (Nein) und RETURN (identisch mit "Nein"). ASK setzt dann den Fehlercode mit 5 (=WARN), wenn mit "Ja" (Eingabe von Y) geantwortet wurde. Wurde als Antwort "N" eingegeben oder die Taste RETURN gedrückt, wird der Fehlercode mit 0 gesetzt. Dieser Befehl ist normalerweise nur in Befehlsdateien hilfreich.

Siehe auch: IF, EXECUTE

ASSIGN

Format: ASSIGN [[<Name>:] <Verzeichnis>] [LIST] [EXISTS] [REMOVE]

Schablone: ASSIGN "NAME,VERZEICHNIS,LIST/S,EXISTS/S, REMOVE/S"

Zweck: Zuweisung eines logischen Gerätenamens an ein Verzeichnis des Dateisystems.

Pfad: C:ASSIGN

Bedeutung:

Bei der Version 1.3 druckt ASSIGN nun Gerätenamen mit mehr als drei Zeichen korrekt. Wird zusammen mit einem logischen Gerätenamen das Schlüsselwort LIST angegeben, sucht ASSIGN in der Zuweisungsliste nach diesem Namen und entfernt ihn.

Wird zusammen mit einem logischen Gerätenamen das Schlüsselwort EXISTS angegeben, sucht ASSIGN in der Zuweisungsliste nach diesem Namen und zeigt das Gerät und das Verzeichnis an, das dem Gerät zugewiesen ist. Wird das Gerät nicht gefunden, wird der Fehlercode mit WARN gesetzt.

Die Option REMOVE trennt einen Datenträger oder ein Gerät von der Liste der momentan zugewiesenen Geräte. Sie gibt keine Ressourcen frei, sondern entfernt nur den Namen aus der Liste. DIESE OPTION WIRD HAUPTSÄCHLICH BEI DER SOFTWAREENTWICKLUNG VERWENDET. DER UNACHTSAME EINSATZ DIESER OPTION KANN EINEN GURU-FEHLER VERURSACHEN.

Beispiele:

```
1> ASSIGN fonts: EXISTS
FONTS: Workbench1.3D:fonts

1> ASSIGN fonts: LIST
Volumes:
RAM Disk [Mounted]

Workbench1.3D [Mounted]

Directories:
CLIPS:  RAM DISK:clipboards
ENV:    RAM DISK:env
T:      RAMD DISK:t
S:      Workbench1.3D:s
L:      Workbench1.3D:l
C:      Workbench1.3D:c
DEVS:   Workbench1.3D:devs
LIBS:   Workbench1.3D:libs
SYS:    Workbench1.3D:

Devices:
SPEAK NEWCON DF2 DF0 PRT
PAR SER RAW CON RAM
```

Beachten Sie, daß die Zuweisung "FONTS: Workbench1.3D:fonts" aus der Zuweisungsliste entfernt wurde.

AVAIL

Format: AVAIL [CHIP|FAST|TOTAL]

Schablone: AVAIL "CHIP/S,FAST/S,TOTAL/S"

Zweck: Zeigt den verfügbaren CHIP- und FAST-Speicher an.

Pfad: C:AVAIL

Bedeutung:

Der Befehl AVAIL lieferte Ihnen eine zusammenfassende Aufstellung des RAM-Speichers im System, wobei CHIP- wie auch FAST-Speicher berücksichtigt wird. Für jeden Speichertyp gibt AVAIL den Gesamtumfang, den momentan verfügbaren und momentan belegten Bereich sowie den größten zusammenhängenden Speicherblock an, der noch nicht zugewiesen ist.

Bei Verwendung der Optionen CHIP, FAST und/oder TOTAL zeigt AVAIL anstelle der vollständigen Aufstellung nur die Anzahl der verfügbaren Bytes des CHIP-, FAST- oder Gesamt-RAM-Speichers an. Dieser Wert kann für Vergleiche in Befehlsdateien verwendet werden.

Beispiele:

```
1> AVAIL
Type      Available   In-Use     Maximum    Largest
chip      233592      282272     515864     76792
fast      341384      182896     524280     197360
total     574976      465168     1040144    197360

1> AVAIL CHIP
233592
```

Siehe auch: EXECUTE, IF

BINDDRIVERS

Format: BINDDRIVERS

Schablone: BINDDRIVERS

Zweck: Zuordnung der Gerätetreiber zur Hardware des Systems.

Pfad: C:BINDDRIVERS

Bedeutung:

Der Befehl BINDDRIVERS ist normalerweise ein Teil der Startsequenz. Er wird verwendet, um Gerätetreiber, die im Verzeichnis SYS:Expansion gefunden werden, Hardware-Erweiterungen zuzuordnen, die durch die Erweiterungsbibliothek automatisch konfiguriert werden. Das bedeutet, daß, falls sich Treiber für Hardware-Erweiterungen in der Schublade Expansion befinden, die Hardware beim Neustart des Systems automatisch konfiguriert wird.

CHANGETASKPRI

Format: CHANGETASKPRI <Priorität> [<Prozeß>]

Schablone: CHANGETASKPRI "PRIORITÄT/A,PROZESS/K"

Zweck: Änderung der Priorität von Prozessen, die über den Befehlsprozessor gestartet wurden.

Pfad: C:CHANGETASKPRI

Bedeutung:

Da der Amiga ein Multitasking-System (Verarbeitung mehrerer Aufgaben, engl. Tasks, gleichzeitig) ist, verwendet es Prioritätsnummern, um festzugelegen, welche Tasks es in welcher Reihenfolge auszuführen hat. Gewöhnlich weisen die meisten Tasks die Priorität 0 auf, und die Zeit- und Anweisungszyklen werden gleichmäßig zwischen ihnen aufgeteilt. Durch CHANGETASKRPI kann die Priorität des angegebenen Prozesses geändert werden. (Wird kein Prozeß angegeben, bezieht sich der Befehl auf den momentan ablaufenden.) Alle Tasks, die von <Prozeß> gestartet werden, übernehmen dessen Priorität.

Der zulässige Wertebereich für <Priorität> liegt zwischen -128 und 127. Es sollten jedoch keine höheren Werte als +5 angegeben werden, um die Unterbrechung von wichtigen Systemtasks zu vermeiden.

Die Prozeßnummern der Befehlsprozessor-Tasks können unter Verwendung des Befehls STATUS ermittelt werden.

Beispiel:

```
1> CHANGETASKPRI 5
```

Die Priorität der aktuellen Befehlsprozessor-Task wird zu 5 gesetzt. Alle Tasks, die von dieser Befehlsprozessor-Sitzung gestartet werden, weisen ebenfalls die Priorität 5 auf. Sie haben Vorrang vor allen anderen Benutzertasks, die ohne den Befehl CHANGETASKPRI erstellt wurden (und die Priorität 0 aufweisen).

Siehe auch: STATUS

COPY

Format: COPY [[FROM] <Name>] [TO] <Name> [ALL] [QUIET] [BUF|BUFFER = <nn>][CLONE] [DATE] [COM] [NOPRO]

Schablone: COPY "FROM, TO/A,ALL/S,QUIET/S,BUF=BUFFER/K, CLONE/S,DATE/S,COM/S,NOPRO/S"

Zweck: Kopieren einer Datei oder eines Verzeichnisses.

Pfad: C:COPY

Bedeutung:

Der Befehl COPY kann verwendet werden, um mehrere Dateien zu einem Gerät zu kopieren. Verwenden Sie entweder die Option ALL, um ein gesamtes Verzeichnis incl. seiner Unterverzeichnisse zu kopieren oder geben Sie Jokerzeichen für die Mustererkennung an, um alle Dateien zu kopieren, deren Namen eine gemeinsame Folge von Zeichen oder Symbolen aufweisen.

Sie können z. B. alle Dateien eines Verzeichnisses kopieren, indem Sie den Verzeichnisnamen für die Mustererkennung angeben. (Die vollständige Erklärung

der Mustererkennung ist im AmigaDOS-Handbuch zu finden. Hinweis: Die Beschränkung auf 31 Zeichen Namenslänge bei Verwendung von Jokerzeichen wurde aufgehoben).

Die Option BUF= wird verwendet, um die Anzahl der Puffer mit 512 Byte Kapazität festzulegen, die während des Kopiervorgangs benutzt werden. (Standardwert ist 100 KByte oder 200 Puffer.) Es ist oft sinnvoll, die Anzahl der Puffer zu beschränken, wenn zur RAM-Disk kopiert wird.

Die Option CLONE weist den Befehl COPY an, das Datum, Kommentare und Schutzbits der Originaldatei mit in die Zieldatei zu übernehmen. Die Optionen DATE und COM bewirken, daß der Befehl COPY das Datum bzw. Kommentare der Originaldatei mitkopiert. Normalerweise übernimmt der Befehl COPY beim Erstellen der Kopie die Schutzbits der Originaldatei. Durch die Option NOPRO kann das verhindert werden.

Wenn bei früheren Versionen ein Verzeichnis als Ziel angegeben wurde, das nicht existierte, wurde vom Befehl COPY kein Verzeichnis erstellt. Bei der Version 1.3 erstellt COPY ein neues Zielverzeichnis. Sie können auch ein Paar von Anführungszeichen (" ") verwenden, um sich bei der Angabe des Ziels auf das aktuelle Verzeichnis zu beziehen.

DATE

Format: DATE [<Datum>] [<Zeit>] [TO=VER<Dateiname>]

Schablone: DATE "DATE,TIME,TO=VER/K"

Zweck: Anzeige oder Setzen des Datums und/oder der Zeit im System.

Pfad: C:DATE

Bedeutung:

Beim Befehl DATE ist es nun möglich, bei der Datumseingabe den Wert mit einer führenden Null oder als einzelne Ziffer einzugeben. Bei früheren Versionen mußte eine führende Null angegeben werden.

DELETE

Format: DELETE <Name>* [ALL] [Q|QUIET]

Schablone: DELETE "NAME(N),ALL/S,Q=QUIET/S"

Zweck: Löschen von bis zu 10 Dateien oder Verzeichnissen.

Pfad: C:DELETE

Bedeutung:

Für das Löschen von Dateien können Jokerzeichen für die Mustererkennung verwendet werden. Die Beschränkung auf 31 Zeichen bei Verwendung von Jokerzeichen wurde aufgehoben. (Die vollständige Erklärung der Mustererkennung ist im AmigaDOS-Handbuch zu finden.)

DIR

Format: DIR [<Name>] [OPT A|I|AI|D] [ALL] [DIRS] [INTER] [FILES]

Schablone: DIR "NAME,OPT/K,ALL/S,DIRS/S,INTER/S,FILES/S"

Zweck: Anzeige einer sortierten Liste der Dateien in einem Verzeichnis.

Pfad: C:DIR

Bedeutung:

ALL, DIRS und INTER sind nun Synonyme für die Optionen A, D bzw. I. Außerdem gibt es nun eine Option COMMAND= für den interaktiven Modus (Aufruf durch OPT I oder INTER), die die Ausführung fast jedes AmigaDOS-Befehls während der interaktiven Verzeichnisauflistung erlaubt.

Wenn Sie im interaktiven Modus einen allgemeinen Befehl verwenden wollen, geben Sie bei der Eingabeaufforderung "?" ein C (oder COM) ein. DIR fragt Sie dann nach dem Befehl. Geben Sie den gewünschten Befehl ein und drücken Sie RETURN. Der Befehl wird ausgeführt und DIR fortgesetzt. Sie können auch das C und den Befehl in einer Zeile zusammenfassen, indem Sie den Befehl nach C in Anführungszeichen schreiben. Zum Beispiel C "type prefs.info hex".

Die Option FILES bewirkt, daß nur die Dateien im Verzeichnis aufgelistet werden. Unterverzeichnisse werden nicht angezeigt.

DISKCHANGE

Format: DISKCHANGE <Laufwerk>

Schablone: DISKCHANGE "LAUFWERK/A"

Zweck: Mitteilung an den Amiga, daß in einem 5-1/4-Zoll-Laufwerk eine *Diskette* gewechselt wurde.

Pfad: C:DISKCHANGE

Bedeutung:

Der Befehl DISKCHANGE muß verwendet werden, wenn Sie 5-1/4-Zoll-Laufwerke mit Ihrem Amiga einsetzen. Bei jedem Diskettenwechsel in diesen Laufwerken muß das System durch DISKCHANGE über die Änderung informiert werden.

Der Befehl ist auch nützlich, um dem Arbeitstisch mitzuteilen, daß mittels des Befehls RELABEL der Name einer Diskette geändert wurde.

Beispiel:

Wenn ein Kommunikationsfenster erscheint und Sie dazu auffordert, eine neue Diskette in das 5-1/4-Zoll-Laufwerk einzulegen (df2: genannt), müssen Sie die Diskette einlegen und dann den Befehl

```
1> DISKCHANGE df2:
```

eingeben. AmigaDOS erkennt dann die neue Diskette, und Sie können mit der Arbeit fortfahren.

DISKDOCTOR

Format: DISKDOCTOR <Laufwerk>

Schablone: DISKDOCTOR "LAUFWERK/A"

Zweck: Versucht, Fehler auf der angegebenen Diskette zu beseitigen.

Pfad: C:DISKDOCTOR

Bedeutung:

DISKDOCTOR versucht, eine fehlerhafte Diskette soweit wiederherzustellen, daß die darauf enthaltenenen Dateien gerettet und auf eine einwandfreie Diskette kopiert werden können. Wenn AmigaDOS feststellt, daß eine Diskette fehlerhaft ist, erscheint ein Kommunikationsfenster, das angibt, daß die Diskette nicht für gültig erklärt werden kann. Unter Verwendung von DISKDOCTOR können Sie versuchen, die Dateistruktur der Diskette wiederherzustellen. Nach der Ausführung von DISKDOCTOR müssen Sie die Dateien auf eine andere Diskette kopieren (mit COPY, nicht DISKCOPY!) und die fehlerhafte dann neu formatieren.

Sie können DISKDOCTOR mit dem Standard-Dateisystem und mit FastFileSystem verwenden. Wird jedoch DISKDOCTOR mit FastFileSystem eingesetzt, MÜSSEN SIE SICHERSTELLEN, DASS DAS SCHLÜSSELWORT DOSTYPE IN DER DATEI MOUNTLIST ALS 0X444F5301 GESETZT IST. Verwenden Sie den Befehl DISKDOCTOR nie für eine FastFileSystem-Partition, wenn das Schlüsselwort DosType nicht korrekt gesetzt ist. (Weitere Informationen über FastFileSystem entnehmen Sie bitte dem Anhang A.)

Bei der Version 1.3 prüft der Befehl DISKDOCTOR, ob ausreichend Speicher vorhanden ist und ändert den Boot-Block zum Typ DOS.

Beispiel:

Wenn Sie die Meldung "`Volume Workbench is not validated`" oder "`Error validating disk/Disk ist unreadable`" erhalten, können Sie den Befehl DISKDOCTOR verwenden, um die Dateien auf der Diskette zu retten. Befindet sich die fehlerhafte Diskette z. B. im Laufwerk df1:, geben Sie ein:

```
1> DISKDOCTOR df1:
```

AmigaDOS fordert Sie dann auf, die zu korrigierende Diskette einzulegen und die Taste RETURN zu drücken. DISKDOCTOR liest dann jeden Zylinder auf der Diskette. Wird dabei ein Fehler festgestellt, erscheint die Meldung "`Hard error Track xx, Surface xx`". Während jede Datei und jedes Verzeichnis ersetzt wird, wird der Name am Bildschirm angezeigt. Nach der Beendigung von DISKDOCTOR erscheint die Meldung: "`Now copy files required to a new disk and reformat this disk.`" (Jetzt alle Dateien auf eine neue Diskette kopieren und die alte neu formatieren.)

ECHO

Format: ECHO <Zeichenfolge> [NOLINE] [FIRST <nn>] [LEN <nn>]

Schablone: ECHO "STRING, NOLINE/S,FIRST/S,LEN/S"

Zweck: Anzeige einer Zeichenfolge.

Pfad: C:ECHO

Bedeutung:

Neu bei Version 1.3 ist die Option NOLINE. Wird diese Option angegeben, beginnt ECHO nach der Ausgabe der Zeichenfolge nicht automatisch eine neue Zeile.

Die Optionen FIRST und LEN ermöglichen die Ausgabe eines Teils der Zeichenfolge. Durch FIRST <nn> wird die Zeichenposition angegeben, bei der die Ausgabe beginnt. Mit LEN <nn> wird die Anzahl der Zeichen festgelegt, die der auszugebene Teil der Zeichenkette enthalten soll, beginnend mit dem ersten Zeichen.

Wird die Option FIRST nicht angegeben, sondern nur das Schlüsselwort LEN, besteht die ausgegebene Zeichenfolge aus den am weitesten rechts stehenden <nn> Zeichen der Hauptzeichenfolge. Ist z. B. Ihre Zeichenfolge 20 Zeichen lang und es wird LEN 4 angegeben, werden das 17., 18., 19. und das 20. Zeichen der Hauptzeichenfolge ausgegeben.

Die auszugende Zeichenfolge kann auch Steuerzeichen enthalten, die folgendermaßen einzusetzen sind:

*N bewirkt Übergang zu einer neuen Zeile.

*E steht für das Escape-Zeichen (Dezimalcode 27). Die im Anhang C des Benutzerhandbuchs aufgelisteten ANSI-Steuersequenzen stehen so auch in CLI und SHELL zur Verfügung, soweit sie auf dem Bildschirm ausführbar sind. Beispiel: *E[1m schaltet auf Fettschrift und *E[0m zurück auf Normalschrift.

Sollen die Zeichenfolgen *N bzw. *E jedoch normal gedruckt und nicht als Steuersequenzen interpretiert werden, ist ihnen ein weiteres Sternchen voranzustellen: **N, **E.

ENDCLI

Format: ENDCLI

Schablone: ENDCLI

Zweck: Beendigung eines interaktiven CLI- oder SHELL-Prozesses.

Pfad: C:ENDCLI

Bedeutung:
ENDCLI kann verwendet werden, um einen CLI- oder einen SHELL-Prozeß zu beenden.

ENDSKIP

Format: ENDSKIP

Schablone: ENDSKIP

Zweck: Abschluß eines SKIP-Blocks in einer Befehlsdatei.

Pfad: C:ENDSKIP

Bedeutung:

Wird während der Verarbeitung eines SKIP-Blocks der Befehl ENDSKIP gefunden, wird die Verarbeitung der Befehlsdatei mit der Zeile fortgesetzt, die auf den Befehl ENDSKIP folgt. (Der Fehlercode wird auf WARN gesetzt.)

EVAL

Format: EVAL <Wert1> <Operation> <Wert2> [TO <Datei>] [LFORMAT=<Zeichenfolge>]

Schablone: EVAL "WERT1,OPERATION,WERT2,TO/K,LFORMAT/K"

Zweck: Ermittlung des Wertes einfacher Ausdrücke.

Pfad: C:EVAL

Bedeutung:

Der Befehl EVAL wird verwendet, um die Ergebnisse einfacher Ganzzahl-Ausdrücke (mit 1 und 2 Argumenten) zu ermitteln und auszugeben. <Wert1> und <Wert2> können als Dezimal-, Hexadezimal- oder Oktalzahlen angegeben werden. Standardgemäß werden Dezimalzahlen angenommen. Hexadezimalzahlen werden durch führende 0X oder #X gekennzeichnet. Oktalzahlen werden mit führender 0 (Null) oder einem führenden Zeichen # angegeben (gefolgt von weiteren Ziffern). Alphabetische Zeichen werden durch eine einzelnes Hochkomma (' , Tastenkombination Alt-Ä) gekennzeichnet.

Die unterstützten Operationen und die entsprechenden Symbole lauten wie folgt:

Addition	+	Nicht	~
Subtraktion	-	Verschiebung nach links	<<
Multiplikation	*	Verschiebung nach rechts	>>
Division	/	Negation	-
Modulo	mod	Exklusiv Oder	xor
Und	&	Bitweise Identität	eqv
Oder	|		

Die Ausgabe erfolgt standardgemäß als Dezimalzahl, durch das Schlüsselwort LFORMAT kann jedoch ein anderes Format gewählt werden. In Verbindung mit dem Schlüsselwort LFORMAT wird die Formatierung für den Ausdruck des Ergebnisses festgelegt. Es können die Kennzeichnungen %X (hexadezimal), %O (oktal), %N (dezimal) oder %C (Zeichen) verwendet werden. Beachten Sie, daß bei den Optionen %X und %O die Anzahl der ausgegebenen Stellen angegeben werden muß (d. h. %X8 ergibt 8 Stellen einer Hexadezimalausgabe).

Bei Verwendung von LFORMAT kann auch der Beginn einer neuen Zeile nach dem Druck des Ergebnisses gefordert werden, indem Sie *N in der Befehlsfolge aufführen.

EVAL kann in Befehlsdateien als Zähler für Schleifen verwendet werden. In diesem Fall ist die Option TO, welche die Ausgabe von EVAL zu einer Datei sendet, sehr hilfreich.

Beispiele:

```
1>  EVAL  4  *  -5
-20

1>  EVAL  0x4f  / 010  LFORMAT="Das Ergebnis ist %X4*N")
Das Ergebnis ist 9

Beispiel in einer Befehlsdatei:
.key  loop/a
;  Beispiel einer Schleife unter Verwendung von EVAL und SKIP
.bra  {
.ket  }
echo  >env:loop  {loop}
lab  start
echo  "Schleifen-Nr."  noline
type  env:loop
eval  <env:loop >NIL:  to=t:qwe  {$$}  value2=1  op=-  ?
type  >env:loop  t:qwe{$$}
If  val  $loop  GT  0
skip  start  back
endif
echo  "Beendet"
```

EXECUTE

Format: EXECUTE <Befehlsdatei> <Argumente>

Schablone: EXECUTE "BEFEHLSDATEI,ARGUMENTE"

Zweck: Ausführung einer Befehlsdatei mit dem Ersetzen von Argumenten.

Pfad: C:EXECUTE

Bedeutung:

EXECUTE verwendet für Zwischenspeicherungen nun den logischen Namen T:, falls er über ASSIGN zugewiesen wurde. Andernfalls wird das Verzeichnis :T auf demselben Laufwerk benutzt.

Da in Befehlsdateien der Befehl EXECUTE verwendet werden kann, ist er ausgezeichnet dazu geeignet, während der Benutzung der SHELL resident geladen zu werden.

Auf die aktuelle Befehlsprozessor-Nummer kann durch die Zeichen <$$> Bezug genommen werden. Dies ist hilfreich beim Erstellen eindeutiger temporärer Dateien, logischer Zuweisungen und PIPE-Namen.

Siehe auch: ASSIGN und Abschnitt SHELL-SEG in Kapitel 3.

FF

Format: FF [-0] [-N]

Schablone: FF "-0/S,-N/S"

Zweck: Erhöhung der Geschwindigkeit des Amiga bei Textausgabe.

Pfad: C:FF

Bedeutung:

FF (FastFonts) ist ein von Charlie Heath von Firma Microsmiths, Inc. geschriebenes Programm. FF beschleunigt die Ausgabe von Text beim Amiga. Sollen die Routinen FastFonts aktiviert werden, ist die Option -0 zu verwenden. Sollen die Routinen

FastFonts deaktiviert werden, ist die Option -N anzugeben.

FF wird mit Genehmigung der Microsmiths Inc. verwendet.

FILENOTE

Format: FILENOTE [FILE] <Dateiname> COMMENT <Kommentar>

Schablone: FILENOTE "FILE/A,COMMENT/K"

Zweck: Hinzufügen eines Kommentars zu einer Datei.

Pfad: C:FILENOTE

Bedeutung:

Die Länge des Kommentars ist auf maximal 79 Zeichen beschränkt.

GETENV

Format: GETENV <Name>

Schablone: GETENV "NAME/A"

Zweck: Abruf des Werts einer Umgebungsvariablen.

Pfad: C:GETENV

Bedeutung:

GETENV wird verwendet, um den Wert einer Umgebungsvariablen abzurufen. Umgebungsvariablen werden im ENV:-Handler gespeichert. (Momentan wird dieser unter Verwendung der RAM:-Disk simuliert.)

Sie können viele AmigaDOS-Befehle mit dem ENV:-Handler verwenden, wie z. B. DIR, LIST, ASSIGN, etc. Dies verschafft dem Benutzer große Flexibilität beim Arbeiten mit Umgebungsvariablen.

Siehe auch: SETENV

ICONX

Format: ICONX

Schablone: ICONX

Zweck: Ausführung einer AmigaDOS-Befehlsdatei durch Start vom Arbeitstisch (Piktogramm anklicken).

Pfad: C:ICONX

Bedeutung:

ICONX erlaubt Ihnen, Befehlsdateien, die aus AmigaDOS-Befehlen bestehen, vom Arbeitstisch ausgehend ausführen zu lassen. ICONX wechselt zu dem Verzeichnis als aktuellem Verzeichnis, das das Projekt-Piktogramm enthält, bevor es die Befehlsdatei ausführt. Auf dem Arbeitstisch wird ein Ein-/Ausgabefenster geöffnet. Sie können die TOOL TYPES des Piktogramms verwenden, um die Größe des Fensters festzulegen (WINDOW=). Durch die Angabe von DELAY= in den TOOL TYPES wird eine kurze Pause nach der Abarbeitung der Datei bewirkt, die es Ihnen ermöglicht, die Ausgabe zu lesen. Wird eine Null für DELAY= angegeben, wartet ICONX auf Control-C als Befehl für das Beenden.

Erstellen Sie für die Verwendung von ICONX eine Befehlsdatei (eine ASCII-Datei, die AmigaDOS-Befehle enthält) und versehen Sie sie mit einem Projekt-Piktogramm. (Zu diesem Zweck kann NotePad verwendet werden, sofern keine Schriftarten und unterschiedliche Zeichensätze in Ihrem Dokument benutzt werden.) Unter Verwendung von INFO aus dem Workbench-Menü ändern Sie das DEFAULT TOOL des Piktogramms zu c:ICONX. Fügen Sie, sofern gewünscht, die TOOL TYPES WINDOW= und DELAY= hinzu. Speichern Sie dann die geänderten Informationen mit SAVE. Durch Doppelklicken auf das Piktogramm wird die Befehlsdatei ausgeführt.

Sie können mit dem Arbeitstisch arbeiten und die erweiterte Auswahl verwenden, um Arbeitstisch-Dateien an die Befehlsdatei zu übergeben. Diese Dateien werden der Befehlsdatei als Argumente übergeben. Für die Nutzung dieser Möglichkeit muß am Beginn der Befehlsdatei das Schlüsselwort .key eingegeben werden. In diesem Fall wird der AmigaDOS-Befehl EXECUTE verwendet, um die Befehlsdatei auszuführen. Das Hinzufügen von .key "" oder .<Leerschritt> erlaubt den Einsatz von Befehlen, die eine Eingabe in eine ICONX-Befehlsdatei verlangen. (Weitere Informationen über die Ausführung von Befehlsdateien sind im AmigaDOS-Handbuch zu finden.)

Siehe auch: EXECUTE

IF

Format: IF [NOT] [WARN] [ERROR] [FAIL] [<Zeichenfolge> EQ|GT|GE <Zeichenfolge>][VAL][EXISTS<Datei>]

Schablone: IF "NOT/S,WARN/S,ERROR/S,FAIL/S,,EQ/K,GT/K,GE/K, VAL/S,EXISTS/K"

Zweck: Verarbeitung von bedingten Operationen in einer Befehlsdatei.

Pfad: C:IF

Bedeutung:

In einer Befehlsdatei vom Typ EXECUTE führt IF alle nachstehenden Befehle aus, bis ein Befehl ENDIF oder ELSE gefunden wird.

IF unterstützt nun Vergleiche der Art GT (größer als) und GE (größer oder gleich). Normalerweise werden Vergleiche als Vergleiche von Zeichenfolgen durchgeführt. Wird jedoch die Option VAL angegeben, erfolgt ein numerischer Vergleich.

Der Befehl IF ist in der Lage, Umgebungsvariablen zu vergleichen. Soll eine Umgebungsvariable angegeben werden, ist vor dem Namen der Variablen das Zeichen $ einzugeben.

HINWEIS: Es können die Ausdrücke NOT GE für "kleiner als" und NOT GT für "kleiner oder gleich" verwendet werden.

Weitere Informationen über den Befehl IF sind im AmigaDOS-Handbuch zu finden.

INFO

Format: INFO [<Gerät>]

Schablone: INFO "GERÄT"

Zweck: Abruf von Informationen über das/die Datei-System(e).

Pfad: C:INFO

Bedeutung:

Die Ausgabe des Befehl INFO wird nun so ausgerichtet, daß längere Datenträgernamen möglich sind. INFO liest reservierte Blöcke aus dem Umgebungs-Vektor des angegebenen Geräts.

Durch die optionelle Angabe eines Gerätenamens ist INFO in der Lage, die Informationen nur eines Geräts oder Datenträgers auszugeben.

INSTALL

Format: INSTALL LAUFWERK <DF0|DF1|DF2|DF3>: [NOBOOT] [CHECK]

Schablone: INSTALL "LAUFWERK/A,NOBOOT/S,CHECK/S"

Zweck: Veränderung des Boot-Blocks auf einer formatierten Diskette.

Pfad: C:INSTALL

Bedeutung:

INSTALL löscht nun den Speicher, der für den Aufbau des Boot-Blocks verwendet wird. Durch die Option NOBOOT wird die Diskette zur DOS-Diskette, mit der jedoch kein Neustart des Systems erfolgen kann. Durch die Option CHECK wird der Block mit dem gültigen Neustart-Code verglichen. INSTALL gibt an, ob eine Diskette für den Neustart verwendbar ist oder nicht, und ob sich der standardgemäße Neustart-Code von Commodore-Amiga auf der Diskette befindet. Der Fehlercode wird auf 0 gesetzt, wenn der Neustart-Code dem Standard entspricht. INSTALL CHECK liefert den Fehlercode 5, wenn der Boot-Block keinen Neustart-Code enthält, und den Fehlercode 10, wenn ein (virusverdächtiger) Nicht-Standard-Boot-Block vorgefunden wird.

HINWEIS: Die Option NOBOOT des INSTALL-Befehls schreibt einen Boot-Block ohne Neustart-Code auch auf eine Diskette, die keine DOS-Diskette ist.

JOIN

Format: JOIN <Name> <Name>* AS|TO <Name>

Schablone: JOIN "NAME(N),TO=AS/K"

Zweck: Bildung einer neuen Datei aus bis zu 15 Ausgangsdateien.

Pfad: C:JOIN

Bedeutung:

TO ist nun ein Synonym für AS.

LIST

Format: LIST [<Verzeichnis|Muster>] [P|PAT <Muster>] [KEYS] [DATES] [NODATES] [TO <Dateiname>] [SUB <Subname>] [SINCE <Datum>] [UPTO <Datum>] [QUICK] [BLOCK] [NOHEAD] [FILES] [DIRS] [LFORMAT <Zeichenfolge>]

Schablone: LIST "DIR.P=PAT/K,KEYS/S,DATES/S,NODATES/S, TO/K,SUB/K,SINCE/K,UPTO/K,QUICK/S,BLOCKS/S, NOHEAD/S,FILES/S,DIRS/S,LFORMAT/K"

Zweck: Anzeige der gewünschten Informationen über Verzeichnisse und Dateien

Pfad: C:LIST

Bedeutung:

Der Befehl LIST zeigt nun die neuen Schutzbits: s (Script), p (Pure - wird mit dem Befehl RESIDENT verwendet) und a (Archive).

Wird nach einem Muster gesucht, kann innerhalb eines Verzeichnisses eine Zeichenfolge für die Suche angegeben werden. Wenn z., B. nach einem Muster im Verzeichniss C: gesucht werden soll, kann LIST C:l#? eingegeben werden. Dadurch werden alle Dateien im Verzeichnis C: angezeigt, deren Name mit einem "l" beginnt.

Die Optionen für den Befehl LIST lauten wie folgt:

QUICK	Es werden keine zusätzlichen Leerschritte nach jedem Dateinamen ausgegeben.
BLOCK	LIST zeigt die Größen der Dateien in Blöcken statt in Byte.
NOHEAD	Unterdrückt die Ausgabe der Informationen im Vorspann.
FILES	LIST zeigt nur Dateinamen an (keine Verzeichnisse).
DIRS	LIST zeigt nur Verzeichnisse an (keine Dateinamen).

LFORMAT verändert die Ausgabeform des Befehls LIST und kann für die schnelle Erstellung von Befehlsdateien verwendet werden. Wird LFORMAT angegeben, werden die Optionen QUICK und NODATES automatisch gewählt. Bei Verwendung von LFORMAT muß eine Zeichenfolge angegeben werden, durch die das Ausgabeformat festgelegt wird. Diese Zeichenfolge wird mit in die Befehlsdatei übernommen. Außerdem muß, wenn die ausgegebenen Informationen gespeichert werden sollen, durch den Befehl > eine Umleitung zu einer Datei erfolgen.

Das Befehlsformat für die Festlegung einer Zeichenfolge als Ausgabeformat (nachstehend "Zeichenfolge" genannt) lautet LFORMAT="Zeichenfolge". Soll die durch LIST ausgegebene Information in die Zeichenfolge integriert werden, kann sie durch Angabe von %S an der gewünschten Stelle eingesetzt werden. Die Art der Ausgabe, die in die Zeichenfolge eingefügt wird, ist abhängig davon, wie oft %S angegeben wird.

Anzahl von %S	Ausgabe
1	Nur Dateiname
2	Pfad, Dateiname
3	Pfad, Dateiname, Pfad
4	Pfad, Dateiname,Pfad, Dateiname

Beispiel:

```
1> LIST >RAM:qwe #? LFORMAT="protect %S -d"
```

Eine neue Befehlsdatei mit dem Namen "qwe" wird in RAM: erstellt. Der Inhalt wird eine Liste aller Dateien im aktuellen Verzeichnis enthalten. Bei der Ausführung von "qwe" wird der Löschschutz jeder Datei entfernt. Der Inhalt kann z. B. wie folgt lauten:

```
protect Expansion.info -d
protect Trashcan -d
protect .info -d
protect c -d
protect Clock.info -d
protect Prefs -d
protect Clock -d
protect System -d
protect L -d
protect Shell -d
protect devs -d
protect S -d
protect Shell.info -d
protect T -d
protect fonts -d
protect libs -d
protect Empty -d
protect Utilities.info -d
protect Disk.info -d
protect Prefs.info -d
protect System.info -d
protect Empty.info -d
protect Trashcan.info -d
protect Utilities -d
protect Expansion -d
```

LOADWB

Format: LOADWB [DELAY] [-DEBUG]

Schablone: LOADWB "DELAY/S,-DEBUG/S"

Zweck: Start des Arbeitstisches (Workbench).

Pfad: C:LOADWB

Bedeutung:

Der Befehl LOADWB wird verwendet, um den Arbeitstisch zu starten. Normalerweise erfolgt dies nur beim Neustart, indem der Befehl LOADWB in der Datei Startup-Sequence aufgeführt wird.

Der Arbeitstisch stellt fest, welche Pfade aktiv sind, während der Befehl LOADWB ausgeführt wird. Er verwendet diese Pfade für jede CLI- oder SHELL-Sitzung, die vom Arbeitstisch gestartet wird.

Wird die Option DELAY angegeben, wartet der Befehl LOADWB vor der Beendigung drei Sekunden. Dies ermöglicht das Abschließen der Diskettenaktivität, bevor die Ausführung fortgesetzt wird.

Die Option -DEBUG veranlaßt den Arbeitstisch dazu, ein verborgenes Workbench-Menü zu aktivieren. Dieses Menü enthält die Menü-Punkte Debug und FlushLibs. Der Menü-Punkt Debug startet den ROM-residenten Amiga-Debugger ROMWACK. Ihr System muß deshalb, wenn Sie diesen Befehl verwenden wollen, ein serielles Terminal angeschlossen haben, das mit 9.600 Baud (festgelegter Wert) arbeitet! FlushLibs veranlaßt den Arbeitstisch, so viel Speicher wie möglich zuzuweisen, indem alle Bibliotheken, Geräte, Zeichensätze, etc., die sich resident im Speicher befinden, jedoch momentan nicht im Einsatz sind, entfernt werden. (Vorausgesetzt, daß die spezielle Bibliothek oder das spezielle Gerät eine Entfernung zuläßt.)

Die beiden Optionen DELAY und -DEBUG sind nicht gleichzeitig verwendbar. Sie können nur jeweils eine der beiden wählen.

LOCK

Format: LOCK <Laufwerk>: [ON|OFF] [<Paßwort>]

Schablone: LOCK "LAUFWERK/A,ON/S,OFF/S,PASSWORT"

Zweck: Aktivieren des Schreibschutzes einer Festplatte bei Verwendung des FastFileSystem.

Pfad: C:LOCK

Bedeutung:

Der Befehl LOCK aktiviert oder deaktiviert den Schreibschutz eines Festplattentreibers oder einer Partition, die unter Verwendung des FastFileSystem angelegt wurde. Die Verriegelung bleibt wirksam, bis das System neu gestartet oder bis sie durch den Befehl LOCK OFF ausgeschaltet wird. Es kann wahlweise ein Paßwort von 4 Zeichen angegeben werden. Wird dieses Paßwort beim Sperren einer Festplattenpartition verwendet, muß das gleiche Paßwort auch beim Aufheben des Schreibschutzes angegeben werden.

Siehe auch: Abschnitt über das FastFileSystem im Anhang A.

MOUNT

Format: MOUNT <Gerät> [FROM <Datei>]

Schablone: MOUNT "GERÄT/A,FROM/K"

Zweck: Anmeldung eines neuen AmigaDOS-Geräts.

Pfad: C:MOUNT

Bedeutung:

MOUNT wird verwendet, um ein neues AmigaDOS-Gerät anzumelden. Mit anderen Worten, MOUNT informiert AmigaDOS darüber, daß ein zusätzliches Gerät zum System hinzugefügt wurde. Wenn der Befehl MOUNT eingegeben wird, sucht er in der Datei DEVS:MountList (oder der Datei, die wahlweise mittels FROM angegeben wurde) nach den Parametern des Geräts, das angemeldet wurde.

Siehe auch: Abschnitt über MountList im Kapitel 3.

NEWCLI

Format: NEWCLI [<Fensterfestlegungen>] [FROM<Dateiname>]

Schablone: NEWCLI "WINDOW,FROM/K"

Zweck: Start eines neuen interaktiven CLI mit einem neuen Fenster.

Pfad: C:NEWCLI

Bedeutung:

Die Standard-Startdatei S:CLI-STARTUP wird für den Start verwendet, sofern keine andere Datei mittels der Option FROM angegeben wird.

Wenn Sie anstelle eines CLI eine SHELL aufrufen wollen, verwenden Sie den Befehl NEWSHELL.

NEWSHELL

Format: NEWSHELL [<Fensterfestlegungen>] [FROM<Dateiname>]

Schablone: NEWSHELL "WINDOW,FROM/K"

Zweck: Start einer neuen interaktiven SHELL mit einem NEWCON-Fenster.

Pfad: C:NEWSHELL

Bedeutung:

Der Befehl NEWSHELL ruft eine neue, interaktive SHELL unter Verwendung des Fenster-Handlers NEWCON: auf. Damit NEWSHELL korrekt arbeitet, muß NEWCON: zuerst angemeldet (Befehl MOUNT) und SHELL-SEG resident aufgenommen werden. (Dies wird normalerweise in der Standard-Startsequenz der Version 1.3 durchgeführt.)

Wurde SHELL-SEG nicht resident angelegt, ruft NEWSHELL anstelle eines neuen SHELL-Fensters ein neues CLI-Fenster auf. Wurde NEWCON: nicht angemeldet (Befehl MOUNT), verwendet NEWSHELL ein CON:-Fenster.

NEWSHELL verwendet die Standard-Startdatei S:SHELL-STARTUP, sofern nicht mittels FROM ein anderer Dateiname angegeben wird.

PATH

Format: PATH [SHOW] [ADD<Verzeichnis>*] [RESET] [QUIET]

Schablone: PATH "SHOW/S,ADD,RESET/S,QUIET/S"

Zweck: Steuerung der Pfadliste, die der Befehlsprozessor für die Suche nach Befehlen verwendet.

Pfad: C:PATH

Bedeutung:

Der Befehl PATH erlaubt die Anzeige, Erweiterung oder Änderung des Suchpfads, dem AmigaDOS folgt, wenn es nach einem auszuführenden Programm sucht. Wenn Sie den Befehl PATH alleine oder zusammen mit der Option SHOW angeben, wird der aktuelle Pfad angezeigt.

Verwenden Sie PATH zusammen mit der Option ADD, um Verzeichnisnamen anzugeben, die dem Pfad hinzugefügt werden sollen. Durch einen Befehl PATH ADD kann der Suchpfad um bis zu zehn Verzeichnisse erweitert werden. Die Namen der einzelnen Verzeichnisse müssen dabei durch jeweils mindestens einen Leerschritt getrennt sein, wobei die Angabe der Option ADD wahlweise erfolgen kann.

Soll der vorhandene Suchpfad durch einen völlig neuen ersetzt werden, ist die Option RESET zu verwenden, gefolgt von den Namen der Verzeichnisse. Der vorhandene Suchpfad wird dann, mit Ausnahme des aktuellen und des Verzeichnisses SYS:C, gelöscht und durch den neu angegebenen ersetzt.

Wird das Kennzeichen QUIET angegeben, sucht der Befehl PATH (bei der Durchführung der Option SHOW) "still" nach dem Pfad, ohne Kommunikationsfenster für nicht angemeldete Datenträger anzuzeigen. Wurde ein Datenträger nicht durch den Befehl MOUNT angemeldet, wird nur der Name des Datenträgers angezeigt; die Verzeichnisse werden von PATH nicht aufgelistet.

PROMPT

Format: PROMPT <Eingabeaufforderung>

Schablone: PROMPT "EINGABEAUFFORDERUNG"

Zweck: Änderung der Eingabeaufforderung für die aktuelle SHELL.

Pfad: C:PROMPT

Bedeutung:

Durch den Befehl PROMPT kann bei Verwendung der SHELL automatisch das aktuelle Verzeichnis angezeigt werden. Zum Beispiel:

%N> Zeigt nur die Nummer des Befehlsprozessors an.
%N.%S> Zeigt die Nummer des Befehlsprozessors, einen Punkt und das aktuelle Verzeichnis an.
%S.%N> Zeigt das aktuelle Verzeichnis, einen Punkt und die Nummer des Befehlsprozessors an.

Siehe auch: Abschnitt über die SHELL in Kapitel 1 und Abschnitt SHELL-SEG in Kapitel 3.

PROTECT

Format: PROTECT [FILE] <Dateiname> [FLAGS] <+ oder - Statusbits> [ADD] [SUB]

Schablone: PROTECT "FILE/A,FLAGS,ADD/S,SUB/S"

Zweck: Änderung der Schutzbits einer Datei.

Pfad: C:PROTECT

Bedeutung:

Durch den Befehl PROTECT können nun auch die neuen Schutzbits Script (s), Pure (p) und Archive (a) verwendet werden. Außerdem kann PROTECT nicht nur dazu eingesetzt werden, um Dateien mit Schutzbits zu versehen, sondern auch, um sie zu entfernen. Zu diesem Zweck werden die Schlüsselworte ADD und SUB ebenso wie

die speziellen Kennzeichen + und - verwendet. Durch den Befehl LIST können die Schutzbits, mit denen Dateien versehen sind, angezeigt werden.

Beispiele:

```
1> PROTECT c:ED +rw
```

Die Datei ED im Verzeichnis C: wird mit den Schutzbits r (lesbar) und w (schreibbar) versehen.

```
1> PROTECT l:dateiname -e
```

Das Schutzbit e (ausführbar) der Datei "Dateiname" im Verzeichnis L: wird gelöscht.

Siehe auch: LIST

REMRAD

Format: REMRAD

Schablone: REMRAD

Zweck: Entfernen der resetfesten RAM-Disk

Pfad: C:REMRAD

Bedeutung:

Soll die resetfeste RAM-Disk aus dem System entfernt werden, ohne das System auszuschalten, ist der Befehl REMRAD zu verwenden. Der Befehl REMRAD weist die resetfeste RAM-Disk an, alle darin enthaltenen Dateien zu löschen. Auch ihre Kapazität wird vermindert, so daß sie sehr klein wird. Beim nächsten Neustart des Amiga wird dann die resetfeste RAM-Disk aus dem System entfernt.

RESIDENT

Format: RESIDENT <residenter Name> <Datei> [REMOVE] [ADD] [REPLACE] [PURE] [SYSTEM]

Schablone: RESIDENT "NAME, DATEI, REMOVE/S, ADD/S, REPLACE/S, PURE/S,SYSTEM/S"

Zweck: Laden und Hinzufügen von Befehlen zur Liste residenter Befehle.

Pfad: C:RESIDENT

Bedeutung:

RESIDENT wird verwendet, um Befehle zu laden und der residenten Befehlsliste hinzuzufügen, die durch die SHELL gewartet wird. Dies erlaubt, den Befehl auszuführen, ohne ihn vor jeder Ausführung neu in den Speicher zu laden. Dadurch wird bei der Bearbeitung mehrerer Aufgaben gleichzeitig (Multitasking) die Ausführung schneller und weniger Speicher verwendet. Dieser Befehl ist nur bei Verwendung der SHELL verfügbar. Der Unterschied zur Speicherung eines Befehls in der RAM-Disk ist, daß das Programm beim Laden von der RAM-Disk noch einmal (in lauffähiger Form) in den Speicher geschrieben wird, also mehr Speicherplatz beansprucht.

Nur bestimmte Befehle können resident geladen werden. Der Befehl muß durch zwei oder mehr voneinander unabhängige Programme gleichzeitig benutzt und wiederholt ausgeführt werden können. Befehle, die diese Eigenschaften aufweisen, sind im Schutzfeld mit dem Schutzbit p (Pure) versehen. Viele Befehle im Verzeichnis C:, ebenso der Befehl MORE in der Schublade UTILITIES, sind mit der Kennzeichnung PURE versehen und können resident geladen werden. Verfügt ein Befehl nicht über dieses Kennzeichen, kann er wahrscheinlich nicht ohne Risiko resident geladen werden.

Die Option REPLACE ist die Standard-Option und muß nicht explizit angegeben werden. Wird der Befehl RESIDENT ohne Optionen oder mit der Option REPLACE angegeben, listet er die Programme auf, die sich in der residenten Befehlsliste befinden. Wurde ein <residenter Name> angegeben und der Befehl RESIDENT findet ein Programm dieses Namens bereits in der Liste, wird er versuchen, den Befehl zu ersetzen. Das Ersetzen wird nur dann erfolgreich durchgeführt, wenn sich der bereits residente Befehl nicht im Einsatz befindet. Wird <residenter Name> nicht, sondern nur eine <Datei> angegeben, fügt RESIDENT den Dateinamen in die

residente Befehlsliste ein. HINWEIS: Der vollständige Pfad zur Datei muß mit dem Dateinamen angegeben werden, obwohl nur der Dateiname in die Liste aufgenommen wird.

Wurde die Option SYSTEM angegeben, wird der Befehl dem System-Teil der residenten Befehlsliste hinzugefügt. Befehle, die mit der Option SYSTEM in die residente Befehlsliste aufgenommen werden, können nicht mehr entfernt werden. Sollen die SYSTEM-Dateien bei der Auflistung durch RESIDENT erscheinen, muß die Option SYSTEM angegeben werden.

Die Option PURE zwingt den Befehl RESIDENT dazu, Befehle zu laden, die nicht über die Kennzeichnung PURE verfügen (d. h. deren Schutzfeld das Bit "p" nicht enthält). Sie kann verwendet werden, um die Reinheit (d. h. die Eignung für residentes Laden) anderer Befehle und Programme zu prüfen. WENDEN SIE DIESE OPTION MIT VORSICHT AN. Berücksichtigen Sie, daß ein Befehl, der resident geladen werden soll, in der Lage sein muß, durch mehrere Prozesse gleichzeitig verwendet und wiederholt ausgeführt zu werden. Obwohl unwahrscheinlich, ist es möglich, daß einige Ihrer Programme diesen Voraussetzungen vollständig entsprechen. Andere Programme können evtl. nicht durch mehrere Prozesse gleichzeitig verwendet werden, sind jedoch für die wiederholte Ausführung geeignet. Diese Befehle können resident geladen werden, wobei Sie unbedingt darauf achten müssen, daß nicht mehrere Prozesse gleichzeitig darauf zugreifen.

Wenn Sie mit der Option PURE experimentieren wollen, muß sichergestellt sein, daß der Neustart mit einer standardgemäßen, schreibgeschützten Workbench-Diskette erfolgt, die niemals und in keiner Weise verändert wurde. Verwenden Sie keine Verzeichnisse auf einer Festplatte, oder weisen Sie ihr keine Verzeichnisse zu. Öffnen Sie ein SHELL-Fenster und wechseln Sie durch CD zu RAM: oder RAD:. Während des Experimentierens müssen Sie sicherstellen, daß sich alle zu lesenden oder zu schreibenden Dateien in RAM:, RAD: oder auf einer Sicherungs- oder neu formatierten Diskette befinden. Laden Sie dann durch die folgende Eingabe das zu testende Programm resident:

1> RESIDENT Dateiname PURE

Für den Test, ob das Programm wiederholt ausgeführt werden kann, starten Sie es von der SHELL, und verwenden Sie so viele Funktionen des Programms wie möglich. Verlassen Sie dann das Programm. Starten Sie es ein zweites Mal, wobei Sie für den Aufruf, falls möglich, andere Optionen in der Befehlszeile verwenden sollten. Es ist möglich, daß das Programm nun nicht einwandfrei arbeitet, was bedeutet, daß es definitiv nicht für den wiederholten Ablauf geeignet ist. Arbeitet das

Programm korrekt, müssen alle Bestandteile geprüft werden, die es sich normalerweise bei einer Sitzung merkt, z. B. Optionen aus der Befehlszeile, verwendete Teile aus Menüs, Zeichenfolgen für die Suche, Namen der zuletzt geladenen oder gespeicherten Dateien. Sind im Programm noch Informationen aus dem vorhergehenden Ablauf vorhanden, ist es für die wiederholte Ausführung nicht geeignet. Weisen die Programmeinstellungen keine Veränderungen seit dem ersten Ablauf auf, sollten Sie mit dem Test fortfahren und dabei so viele Optionen wie möglich verwenden, um sicherzustellen, daß das Programm auch weiterhin korrekt abläuft. Ist das der Fall, kann es wahrscheinlich als residenter Befehl verwendet werden, **auf den jeweils nur ein Prozeß zugreifen darf.**

Für den Test, ob ein Programm durch mehrere Prozesse gleichzeitig voll genützt werden kann, öffnen Sie eine zweite SHELL, wechseln zu RAM: und versuchen Sie das residente Programm von beiden SHELL-Ebenen gleichzeitig ablaufen zu lassen. Testen Sie alle Programmoptionen, wobei beide laufenden Programme geprüft werden müssen, um sicherzustellen, daß die in einer SHELL gewählten Optionen nicht das Programm in der anderen SHELL beeinflussen. Wenn das Programm Dateien lädt oder speichert, müssen die Optionen für das Laden und Speichern in beiden Programmabläufen gleichzeitig angegeben werden. Überprüfen Sie die geladenen bzw. gespeicherten Dateien, um sicherzustellen, daß sie nicht unvollständig oder fehlerhaft sind. Zu diesem Zeitpunkt kann es sein, daß Sie sicher sind, daß das Programm ohne Risiko resident verwendet werden kann. Sie sollten jedoch in diesem Fall darauf vorbereitet sein, daß Probleme mit Dateien oder Fehler bei der Ausführung auftreten können.

Siehe auch: PROTECT und den Abschnitt SHELL-SEG in Kapitel 3.

RUN

Format: RUN <Befehl> [+ <Kommentar>]*

Schablone: RUN "BEFEHL"

Zweck: Ausführung von Befehlen als Hintergrundprozeß.

Pfad: C:RUN

Bedeutung:

RUN kann nun verwendet werden, um Hintergrundprozesse zu starten, die das Schließen des Befehlsprozessors, über den der Prozeß gestartet wurde, nicht

verhindern. Dazu muß die Ausgabe von RUN unter Verwendung des Symbols > zu NIL: umgeleitet werden. Wenn jedoch das ablaufende Programm mit "*" (dem aktuellen Fenster) fest gekoppelt wird, was aus verschiedenen Gründen geschehen kann, kann auch dieses Fenster nicht geschlossen werden.

Befehle, die über RUN gestartet werden, können aus der Befehlsgruppe stammen, die in der residenten Befehlsliste gespeichert sind. Zur Erhöhung der Geschwindigkeit wird zuerst geprüft, ob der Befehl resident geladen ist, bevor der Befehlspfad für sein Auffinden verwendet wird.

Wird RUN verwendet, um eine neue Sitzung des Befehlsprozessors zu starten, trifft es die nötigen Vorbereitungen, so daß für die neue Sitzung die Standard-Startdatei eingesetzt wird. Die Datei S:SHELL-STARTUP wird benutzt, wenn die SHELL aktiv ist. Anderfalls wird die Datei S:CLI-STARTUP verwendet.

Beispiel:

```
1> RUN >NIL: Beispielprogramm
```

Die Ausgabe vom "Beispielprogramm" wird zu NIL: umgeleitet.

SEARCH

Format: SEARCH FROM <Name> <Muster> [SEARCH] <Zeichenfolge> [ALL] [NONUM] [QUIET] [QUICK] [FILE]

Schablone: SEARCH "FROM, SEARCH/A,ALL/S,NONUM/S, QUIET/S,QUICK/S,FILE/S"

Zweck: Suche nach der angegebenen Zeichefolge in den angegebenen Dateien.

Pfad: C:SEARCH

Bedeutung:

Der Befehl SEARCH liefert nun als Rückmeldung die Ziffer 0, wenn das Objekt gefunden wurde, andernfalls die Ziffer 5 (=WARN). Dadurch wird sein Nutzen für Befehlsdateien erhöht. Durch Eingabe von Control-C wird die Suche beendet.

Die Optionen lauten wie folgt:

NONUM SEARCH druckt keine Zeilennummern mit den Zeichenfolgen.

QUIET SEARCH sucht "still", d. h. keine Ausgaben werden angezeigt.

QUICK SEARCH verwendet ein kompakteres Ausgabeformat

FILE SEARCH sucht eine Datei mit dem angegebenen Namen, statt eine Zeichenfolge in dieser Datei zu suchen.

HINWEIS: Die Begrenzung auf 31 Zeichen bei Verwendung von Jokerzeichen wurde aufgehoben.

SETCLOCK

Format: SETCLOCK LOAD|SAVE|RESET

Schablone: SETCLOCK "LOAD/S,SAVE/S,RESET/S"

Zweck: Setzen oder Abfragen der Echtzeituhr.

Pfad: C:SETCLOCK

Bedeutung:

Dieser Befehl wird verwendet, um die Zeit- und Datumseinstellung der Echtzeituhr (falls vorhanden) mit den Werten der aktuellen Systemzeit festzugelegen (Option SAVE) oder die aktuelle Systemzeit entsprechend der Echtzeituhr zu setzen (Option LOAD). Wird die Option SAVE des Voreinstellerprogramms Preferences verwendet, setzt Preferences die Echtzeituhr und die aktuelle Systemzeit gleichzeitig. Die Option RESET wird benutzt, um die Uhr vollständig zurückzusetzen. Dies kann erforderlich sein, wenn ein fehlerhaftes Programm die Uhr ausschaltet oder die Testbits der Uhr setzt.

SETDATE

Format: SETDATE <Datei> <Datum> <Uhrzeit>

Schablone: SETDATE "DATEI/A, DATUM,UHRZEIT"

Zweck: Änderung des Datums und der Uhrzeit für eine Datei oder ein

Verzeichnis.

Pfad: C:SETDATE

Bedeutung:

Die Regeln für die Eingabe des Befehls SETDATE wurden gelockert, so daß die Ausgabe des Befehls DATE dafür verwendet werden kann. Es ist auch nicht mehr erforderlich, bei der Angabe des Datums eine führende Null einzugeben. SETDATE <Datei> führt eine Veränderung der Dateieinträge durch - der Datums- und Zeiteintrag der Datei wird jeweils entsprechend der aktuellen Systemwerte für Datum und Uhrzeit eingestellt.

Siehe auch: DATE

SETENV

Format: SETENV <Variablenname> <Zeichenfolge der Variablen>

Schablone: SETENV "NAME/A, ZEICHENFOLGE"

Zweck: Setzen einer Umgebungsvariablen.

Pfad: C:SETENV

Bedeutung:

SETENV wird verwendet, um den Wert einer Umgebungsvariablen festzulegen oder zu entfernen. Umgebungsvariablen werden im ENV:-Handler gespeichert (gegenwärtig ist das die RAM:-Disk).

Für das Entfernen der Definition einer Umgebungsvariablen wird der Ausdruck SETENV <Variablenname> ohne Angabe einer Zeichenfolge verwendet. Die Variable bleibt in ENV: vorhanden, jedoch ohne Inhalt.

Beispiele:

1> SETENV Editor Extras1.3D:Tools/MEMACS

Erstellt die Umgebungsvariable "EDITOR", die mit dem Hilfsprogramm MORE benutzt werden kann. Sie gibt an, daß als Editor das Programm MEMACS eingesetzt wird, das sich in der Schublade TOOLS auf der Extras-Diskette befindet.

1> SETENV Editor C:ED

Dieser Befehl verfügt über dieselbe Wirkung, nur daß als Editor der AmigaDOS-Editor ED definiert wird.

Siehe auch: GETENV und Abschnitt über den ENV:-Handler in Kapitel 1.

SETPATCH

Format: SETPATCH [R]

Schablone: SETPATCH "R/S"

Zweck: Korrektur von Fehlern im ROM der Kickstart-Versionen 1.2 und 1.3.

Pfad: C:SETPATCH

Bedeutung:

Die Kickstart-ROMs der Version 1.2 und 1.3 weisen einige bekannte Fehler auf, die durch den Ablauf von SETPATCH unwirksam gemacht werden. SETPATCH muß in der ersten Zeile der Datei Startup-Sequence angegeben sein, so wie das bei der Startdatei der Workbench-Diskette, Version 1.3, der Fall ist. SETPATCH beseitigt die Fehler von Display-Alert(), den mathematischen Ausnahme-Vektoren des Prozessors 68000, der Routine Delete-Layers() der Grafikbibliothek und der Funktion AllocEntry().

Die Option R wird verwendet, um bei Systemen mit 1 MByte CHIP RAM die resetfeste RAM-Disk zu schützen. Dies ist erforderlich, da die Kickstart-Version 1.3 für die Arbeit mit 512 KByte CHIP RAM ausgelegt ist.

Siehe auch: Kapitel 5, Ändern der Startsequenzen.

SKIP

Format: SKIP <MARKE> [BACK]

Schablone: SKIP "MARKE,BACK/S"

Zweck: Sprungbefehl bei der Ausführung von Befehlsdateien.

Pfad: C:SKIP

Bedeutung:

Wird die Option BACK verwendet, beginnt SKIP am Anfang der Datei mit der Suche nach der angegebenen Sprungadresse (Marke). Ohne Angabe der Option BACK beginnt SKIP die Suche in der aktuellen Zeile der Datei. Die Option BACK erlaubt also nun Sprünge nach rückwärts. Nur bis zur letzten zuvor angegebenen Anweisung EXECUTE kann zurückgesprungen werden. Enthält die Datei keine EXECUTE-Anweisungen, ist der Sprung zum Anfang der Datei möglich.

STATUS

Format: STATUS <Prozeß> [FULL] [TCB] [CLI|ALL] [COMMAND]

Schablone: STATUS "PROZESS, FULL/S,TCB/S, CLI=ALL/S, COMMAND/K"

Zweck: Auflistung von Informationen über CLI-/SHELL-Prozesse.

Pfad: C:STATUS

Bedeutung:

STATUS unterstützt nun negative Prioritäten. Mit der neuen Option COMMAND kann STATUS angewiesen werden, nach einem Befehl zu suchen. STATUS durchsucht dann die Liste des Befehlsprozessors nach diesem Befehl. Wird der Befehl gefunden, wird die Nummer des Befehlsprozessors ausgegeben und der Fehlercode mit 0 gesetzt. Andernfalls wird der Fehlercode auf 5 (=WARN) gesetzt.

Dies ist nützlich in Befehlsdateien. Soll zum Beispiel ein BREAK an den Prozeß gesendet werden, der den Befehl ClockPtr ausführt, lautet der Ausdruck:

```
1> STATUS >ram:qwe COMMAND=ClockPtr
BREAK <ram:qwe >NIL: ?
```

Siehe auch: BREAK

TYPE

Format: TYPE <von> [TO <Name>] [OPT H=HEX|N=NUMBER]

Schablone: TYPE "VON/A, TO/S,OPT/K,HEX/S,NUMBER/S"

Zweck: Ausgabe einer Textdatei.

Pfad: C:TYPE

Bedeutung:

TYPE prüft nun, ob die angegebene Zieldatei bereits vorhanden ist. Ist das der Fall, wird eine Meldung ausgegeben, sofern die Option TO nicht explizit eingegeben wird.

Die Optionen H und N sind nun in Form der Schlüsselwörter HEX bzw. NUMBER verfügbar.

VERSION

Format: VERSION <Bibliotheksname|Gerätename> [<Versionsnr.>] [<Revisionsnr.>] [<Einheitennr.>]

Schablone: VERSION "NAME,VERSION, REVISION, EINHEIT"

Zweck: Auffinden der Versions- und Revisionsnummer einer Workbench-Diskette.

Pfad: *C:VERSION*

Bedeutung:

VERSION wird verwendet, um die Versions- und Revisionsnummer einer Bibliothek, eines Geräts oder der Workbench-Diskette zu finden. VERSION kann auch spezielle Versions-/Revisionsnummern suchen und den Fehlercode setzen, falls die gefundenen Versions-/Revisionsnummern höher sind.

Wird VERSION ohne das Argument <Bibliotheksname|Gerätename> angegeben, erfolgt die Ausgabe der Kickstart- und der Workbench-Versionsnummer. Wurde <Bibliotheksname|Gerätename> eingegeben, versucht der Befehl VERSION, die Bibliothek oder das Gerät zu öffnen und die Versions-Informationen zu lesen.

Wurde <Versionsnr.> (und möglicherweise <Revisionsnr.>) angegeben, setzt VERSION den Fehlercode mit 0, falls die gefundenen Versions- und Revisionsnummern höher als die angegebenen Werte oder gleich sind. Andernfalls wird der Fehlercode mit 5 (=WARN) gesetzt. (Wurde keine Revisionsnummer eingegeben, erfolgt kein Vergleich der Revisionsnummern.)

Wird VERSION für die Workbench-Diskette selbst benutzt, gibt der erste Wert die Kickstart-Version, der zweite die Workbench-Version an. Der Befehl VERSION kann nicht zusammen mit einem Pfadnamen verwendet werden, um die Version eines Geräts zu prüfen.

Die Option <Einheitennr.> erlaubt Ihnen, eine Einheitennummer, die ungleich 0 ist, anzugeben. Dies ist beim Zugriff auf Geräte, die aus mehreren Einheiten bestehen (z. B. SCSI-Schnittstelle), gelegentlich erforderlich.

Siehe auch: IF, EXECUTE

WAIT

Format: WAIT <n> [SEC|SECS] [MIN|MINS] [UNTIL <Zeitangabe>]

Schablone: WAIT ",SEC=SECS/S, MIN=MINS/S,UNTIL/K"

Zweck: Einhalten einer Wartezeit der angegeben Länge.

Pfad: C:WAIT

Bedeutung:

WAIT erlaubt bei der Angabe der Wartezeit nun eine führende Null.

Bei der UNTIL-Option muß der Endzeitpunkt im Format SS:MM eingegeben werden, WAIT wartet dann bis zur Mitte dieser Minute.

WHICH

Format: WHICH <Dateiname> [NORES] [RES]

Schablone: WHICH "DATEI/A, NORES/S,RES/S"

Zweck: Absuchen des Befehlspfads nach einem bestimmten Befehl.

Pfad: C:WHICH

Bedeutung:

WHICH erlaubt das Auffinden eines bestimmten Befehls, indem der Ort des Vorhandenseins gemeldet wird. WHICH sucht nach dem angegeben Befehl normalerweise in der residenten Befehlsliste, im aktuellen Verzeichnis, dem/den Befehlspfad(en) und dem Verzeichnis C:. Wird er nicht gefunden, wird der Fehlercode mit WARN gesetzt.

Die Option NORES bewirkt, daß nicht in der residenten Befehlsliste gesucht wird. Durch die Option RES wird nur in der residenten Befehlsliste gesucht.

Beispiele:

```
1>  WHICH  avail
C:avail

1>  WHICH  C:
RAMWB:C
```

2.3 Formate der unveränderten Befehle

BREAK

Format: BREAK <Prozeß> [ALL|C|D|E|F]

Schablone: BREAK "PROZESS/A, ALL/S,C/S,D/S,E/S,F/S"

Zweck: Setzen der Abbruchcodes im angegebenen Prozeß.

Pfad: C:BREAK

CD

Format: CD <Verzeichnis>

Schablone: CD "VERZEICHNIS"

Zweck: Setzen, Ändern oder Anzeigen des aktuellen Verzeichnisses.

Pfad: C:CD

ED

Format: ED [FROM] <Dateiname> [SIZE <n>]

Schablone: ED "FROM/A,SIZE"

Zweck: Edieren einer Textdatei (mit Bildschirmeditor)

Pfad: C:ED

EDIT

Format: EDIT [FROM] <Dateiname> [[TO] <Dateiname>] [WITH <Dateiname>] [VER<Dateiname>] [OPT <Option>]

Schablone: EDIT "FROM/A,TO,WITH/K,VER/K,OPT/K"

Zweck: Sequentielles Edieren einer Textdatei (mit Zeileneditor).

Pfad: C:EDIT

ELSE

Format: ELSE

Schablone: ELSE

Zweck: Alternative bei bedingten Operationen in einer Befehlsdatei.

Pfad: C:ELSE

ENDIF

Format: ENDIF

Schablone: ENDIF

Zweck: Abschluß eines IF-Blocks in einer Befehlsdatei.

Pfad: C:ENDIF

FAILAT

Format: FAILAT <n>

Schablone: FAILAT "FEHLERGRENZWERT"

Zweck: Setzen der Fehlerbedingung einer Befehlsdatei.

Pfad: C:FAILAT

FAULT

Format: FAULT <Fehlernummer>

Schablone: FAULT ", , , , , , , , , ,"

Zweck: Ausgabe der angegebenen Fehlermeldung(en).

Pfad: C:FAULT

LAB

Format: LAB <Zeichenfolge>

Schablone: LAB "ZEICHENFOLGE"

Zweck: Angabe einer Marke (Sprungadresse) in einer Befehlsdatei.

Pfad: C:LAB

MAKEDIR

Format: MAKEDIR <Verzeichnis>

Schablone: MAKEDIR "VERZEICHNIS/A"

Zweck: Erstellen eines neuen Verzeichnisses.

Pfad: C:MAKEDIR

QUIT

FORMAT: QUIT <Rückkehrcode >

Schablone: CD "RÜCKKEHRCODE"

Zweck: Beenden einer Befehlsdatei mit dem angegebenen Rückkehr-, das heißt Fehlercode.

Pfad: C:QUIT

RELABEL

Format: RELABEL DRIVE <Laufwerk>: NAME <Name>

Schablone: RELABEL "DRIVE/A,NAME/A"

Zweck: Ändern des Namens einer Diskette.

Pfad: C:RELABEL

RENAME

Format: RENAME [FROM] <Name> [TO|AS] <Name>

Schablone: RENAME "FROM/A,TO=AS/A"

Zweck: Ändern des Namens einer Datei oder eines Verzeichnisses.

Pfad: C:RENAME

3. Weitere Workbench-Verzeichnisse

Außer den in Kapitel 2 beschriebenen AmigaDOS-Befehlen sind auf der Workbench-Diskette noch viele andere Programme und Dateien enthalten, die Ihnen helfen sollen, die Verarbeitungsleistung zu erhöhen und Ihren Programmieraufwand zu verringern. Das Verzeichnis S: enthält z. B. Befehlsdateien, die es Ihnen erlauben, viele AmigaDOS-Befehle mit der Fähigkeit auszustatten, Jokerzeichen zu akzeptieren. Die Datei MountList im Verzeichnis DEVS: informiert den Amiga über die Merkmale der Peripherie- und Erweiterungsgeräte, mit denen das System ausgestattet werden kann.

In diesem Kapitel sind Dateien beschrieben, die sich in den Verzeichnissen DEVS:, L:, S: und LIBS: der Workbench-Diskette befinden. Einige der Dateien sind neu, während andere für die Version 1.3 geändert wurden. Es wird vorausgesetzt, daß Sie Vorkenntnisse über den Amiga besitzen. Für Neuanwender kann es erforderlich sein, weitere Dokumentationen zu Rate zu ziehen, um die Wirkungsweise dieser Hilfsprogramme vollständig zu verstehen. Erfahrene Benutzer sollten dieses Kapitel gründlich lesen, in dem Neuigkeiten und Änderungen beschrieben sind.

3.1 Geräte (Devices)

Das Verzeichnis DEVS: enthält Dateien ("Gerätetreiber") vom Type .DEVICE, wie z. B. RAMDRIVE.DEVICE, PRINTER.DEVICE, SERIAL.DEVICE, etc. Viele dieser Dateien beziehen sind auf physisch vorhandene Geräte, wie z. B. Peripheriegeräte, die mit den Anschlüssen des Amiga verbunden sind (z. B. die Datei SERIAL.DEVICE steuert die Kommunikation über den seriellen Anschluß, während PARALLEL.DEVICE den Informationsfluß über den parallelen Anschluß, für gewöhnlich zu einem Drucker, steuert.) Dieser Abschnitt befaßt sich nicht mit allen Gerätetreibern. Sie sind in anderen Dokumentationen beschrieben, wie z. B. den ROM Kernel Manuals von Addison-Wesley. Dieser Abschnitt beinhaltet Informationen über die MountList (Liste der angemeldeten Geräte), mit der viele Benutzer vertraut sein müssen, wenn sie ihren Amiga mit Erweiterungsgeräten ausstatten wollen. Außerdem wird die neue Zuweisung CLIPS: beschrieben, die durch das CLIPBOARD.DEVICE verwendet wird.

3.1.1 Liste der Geräte, die angemeldet werden können (MountList)

Die Datei MountList, die im Verzeichnis DEVS: zu finden ist, enthält die Beschreibungen der Geräte, die mittels des AmigaDOS-Befehls MOUNT angemeldet wurden. Es kann erforderlich sein, einen Eintrag für ein Gerät, einen Handler oder ein Dateisystem in die MountList aufzunehmen. Wenn Sie Ihr Amiga-System um ein neues Gerät erweitern, wie z. B. ein Festplattenlaufwerk oder externe Diskettenlaufwerke, muß der Amiga vom Vorhandensein dieser Geräte in Kenntnis gesetzt werden. Dazu wird der Befehl MOUNT verwendet. Der Befehl MOUNT muß jedoch den entsprechenden Eintrag in der Datei MountList lesen, um die Merkmale des Geräts festzustellen.

Die Datei MountList im Verzeichnis DEVS: enthält bereits einige einfache Einträge. Einige davon können unverändert benutzt werden. Es ist jedoch immer empfehlenswert, den Inhalt der Datei äußerst sorgfältig zu prüfen, um sicherzustellen, daß er Ihrem Gerät entspricht.

Ein Eintrag in MountList besteht aus einer Anzahl von Schlüsselwörtern, die das Gerät, den Handler oder das Dateisystem beschreiben, und zu den Schlüsselwörtern gehörenden Werten. Einige Schlüsselwörter beziehen sich nur auf ein Dateisystem oder einen Handler. Ist ein Schlüsselwort nicht angegeben, wird der Standardwert verwendet. Die Standardwerte müssen immer geprüft werden, da sie unter Umständen für das anzumeldende Gerät nicht korrekt sind.

Beim Erstellen eines Eintrags in MountList müssen einige Regeln eingehalten werden:

- Jeder Eintrag in der MountList muß mit dem Namen des Geräts beginnnen.
- Auf Schlüsselwörter muß ein Gleichheitsymbol (=) folgen.
- Schlüsselwörter müssen durch ein Semikolon getrennt oder in jeweils eigenen Zeilen geschrieben werden.
- Kommentare sind erlaubt, sofern sie wie bei der Programmiersprache C geschrieben werden (d. h. Kommentare beginnen mit /* und enden mit */).
- Jeder Eintrag muß mit dem Symbol # enden, das in einer eigenen Zeile geschrieben wird.

Die vom Befehl MOUNT unterstützten Schlüsselwörter lauten:

SCHLÜSSELWORT	FUNKTION
Handler=	Ein Eintrag für einen Handler (z. B. Handler= L:Newcon-Handler)
FileSystem=	Ein Eintrag für ein Dateisystem (z. B. FileSystem= L:FastFileSystem)
Device=	Ein Eintrag für ein Gerät (z. B. Device= ramdrive.device)
Priority=	Die Priorität des Prozesses; der Wert 5 ist für Handler und der Wert 10 für Dateisysteme geeignet.
Unit=	Die Einheitennummer des Geräts.
Flags=	Kennzeichen für OpenDevice (normalerweise 0)
Surfaces=	Die Anzahl der Oberflächen
BlocksPerTrack=	Die Anzahl der Blöcke pro Spur
Reserved=	Die Anzahl der Blöcke für den Boot-Block; muß 2 sein.
PreAlloc=	Die Anzahl der reservierten Blöcke am Ende einer Partition. Wird von einigen Festplattenlaufwerken verwendet, die Informationen in den letzen Blöcken eines Laufwerks speichern. Dieser Wert ist gewöhnlich auf 0 gesetzt, und es ist selten erforderlich, ihn zu ändern. Ziehen Sie bitte die mit dem Festplattenlaufwerk und der Festplattensteuerung gelieferte Dokumentation zu Rate.
Interleave=	Versatz log./phys. Sektoren, geräteabhängig
LowCyl=	Zu verwendender erster Zylinder
HighCyl=	Zu verwendender letzter Zylinder
Stacksize=	Umfang des Stack-Speichers, der dem Prozeß zugewiesen wird.
Buffers=	Anzahl der anfänglich vorhandenen Cache-Puffer
BufMemType=	Speicherart, die für Puffer verwendet wird. (0 und 1 = alle, 2 und 3 = CHIP, 4 und 5 = FAST)
Mount=	Ist der Wert positiv, lädt MOUNT das Gerät oder den Handler sofort. Ist er negativ, wird auf den ersten Zugriff gewartet.
MaxTransfer=	Die maximale Anzahl der übertragenen Blocks; wird mit dem FastFileSystem verwendet
Mask=	Adreßmaske für die Angabe des Speicherbereichs, der für DMA-Übertragungen genutzt wird; wird mit dem FastFileSystem verwendet

GlobVec= Ein globaler Vektor für die Verarbeitung. -1 bedeutet "kein globaler Vektor" (für C- und Assembler-Programme), 0 bewirkt die Bereitstellung eines persönlichen Vektors GV. Ist das Schlüsselwort nicht vorhanden, wird der gemeinsam benutzte globale Vektor verwendet.

Startup= Eine als BPTR auf einen BSTR übertragene Zeichenkette, die beim Start an ein Gerät, einen Handler oder ein Dateisystem übergeben wird.

BootPri= Ein Wert, der die Priorität beim Neustart (Boot) für Geräte angibt, von denen Neustarts durchgeführt und die durch MOUNT angemeldet werden können. Der erlaubte Wertebereich liegt zwischen -129 und 127. Der Wert -129 bedeutet, daß von diesem Gerät kein Neustart durchgeführt werden kann und es nicht automatisch angemeldet wird. Dieser Wert muß für eine resetfeste RAM-Disk verwendet werden, wenn mit dem FastFileSystem gearbeitet wird.

DosType= Gibt die Art des Dateisystems an. Wenn das FastFileSystem verwendet wird, muß das Schlüsselwort DosType mit dem Wert 0x444F5301 angegeben werden. Andernfalls ist der Wert 0x444F5300 festzulegen oder keiner der Werte einzugeben. Künftig werden u. U. noch weitere Werte benutzt.

Die Datei MountList enthält Mustereinträge. Wenn Sie einen neuen MountList-Eintrag erstellen müssen, dann sind für gewöhnlich Anweisungen in der Begleitdokumentation des Geräts vorhanden, das Sie anmelden wollen. Einige MountList-Beispiele sind auch in diesem Kapitel aufgeführt, jeweils zusammen mit der Beschreibung der verschiedenen Handler.

3.1.2 Zwischenspeicher (Clipboard.device)

Eine Neuerung, die bei der Version 1.3 vorgenommen wurde, betrifft den Zwischenspeicher (Clipboard.device). Das logische Gerät CLIPS: erlaubt Ihnen, für das Speichern von Inhalten des Zwischenspeichers ein anderes Verzeichnis anzugeben. In der Startsequenz der Workbench-Diskette wird CLIPS: dem Verzeichnis RAM:clipboards zugewiesen. Falls gewünscht, kann die Zuweisung jedoch geändert werden.

3.2 Tastaturbelegungen (Keymaps)

Keymaps ist ein Unterverzeichnis des Verzeichnisses DEVS: (devs/keymaps). Wie im Abschnitt über SetMap in Kapitel 1 erwähnt, enthält das Verzeichnis Keymaps auf der Workbench-Diskette nur die Tastaturbelegung d. Die internationalen Tastaturbelegungen sind im Verzeichnis Devs/Keymaps auf der Extras-Diskette zu finden:

KEYMAP-DATEI	ENTSPRECHENDE TASTATUR
cdn	Französisch-Kanadische Tastatur
ch1	Französisch-Schweizerische Tastatur
ch2	Deutsch-Schweizerische Tastatur
d	Deutsche Tastatur
dk	Dänische Tastatur
e	Spanische Tastatur
f	Französische Tastatur
gb	Britische Tastatur
i	Italienische Tastatur
is	Isländische Tastatur
n	Norwegische Tastatur
s	Schwedische Tastatur
usa2	Dvorak-Tastatur

Für den Einsatz der gewünschten Tastaturbelegung:

1) Kopieren Sie die Tastaturbelegungsdatei in das Verzeichnis DEVS:keymaps. Zum Beispiel:

```
COPY Extras1.3D:devs/keymaps/d to devs/keymaps
```

2) Verwenden Sie das Programm SetMap (in der Systemschublade), um das System über die Änderung zu informieren.

```
SETMAP  d
```

Wenn Sie eine andere als die Standard-Tastaturbelegung ständig verwenden wollen, kopieren Sie die entsprechende Datei in das Verzeichnis DEVS:keymaps, und ändern Sie unter Verwendung eines Texteditors, wie ED oder MEMACS, die Zuweisung von SetMap in der Startup-Sequence der Workbench-Diskette.

3.3 Zeichensätze (Fonts)

Das Verzeichnis FONTS: der Workbench-Diskette enthält die standardgemäßen Amiga-Zeichensätze: Diamond, Emerald, Garnet, Opal, Ruby, Sapphire und Topaz (normalerweise aktiver Zeichensatz). Ein schneller Weg, die unterschiedlichen Zeichensätze zu sehen, ist, ein Notizblock-Fenster (Notepad) zu öffnen, einen Satz für den Test zu schreiben, und die Zeichensätze aus dem Menü Font auszuwählen. Bei jedem Wählen eines Zeichensatzes ändert sich die Darstellungsweise des Textes im Notizblock-Fenster.

Die Extras-Diskette enthält ebenfalls ein Verzeichnis "Fonts", das drei neue Zeichensätze enthält: Courier, Helvetica und Times. Für den Einsatz der neuen Zeichensätze müssen Sie entweder:

1) Unter Verwendung des AmigaDOS-Befehls COPY einen oder alle Zeichensätze zur Workbench-Diskette kopieren. Zum Beispiel:

```
COPY  Extras1.3D:fonts/Times  to  Workbench1.3D:fonts
```

   Nach dem Kopieren von Zeichensätzen zur Workbench-Diskette muß das Programm FixFonts (in der Systemschublade) verwendet werden, um das Verzeichnis FONTS: zu aktualisieren.

   Oder

2) Durch den Befehl ASSIGN das logische Gerät FONTS: dem Zeichensatzverzeichnis auf der Extras-Diskette zuweisen.

```
ASSIGN  FONTS:  Extras1.3D:fonts
```

Würde nun ein Notizblock-Fenster geöffnet, erschiene nun im Menü zusätzlich Times (nach der Ausführung von Beispiel 1) oder Sie könnten zwischen den Zeichensätzen Helvetica, Courier und Times wählen. Diese Zeichensätze können mit verschiedenen Softwareprodukten, die das Hinzufügen oder Ändern von Zeichensätzen erlauben (z. B. Textverarbeitungsprogrammen) verwendet werden. Beim Einsatz der neuen Zeichensätze muß beachtet werden, daß es sich bei Helvetica und Times um Proportionalschriften und bei Courier um eine Schrift mit identischen Zeichenbreiten handelt.

3.4 Handler (Steuerprogramme)

Im Verzeichnis L: der Workbench-Diskette befinden sich einige neue Handler. Die meisten Handler werden als physische Geräte behandelt und wie ein Diskettenlaufwerk angesprochen. Durch RAM: wird z. B. der RAM-Handler adressiert, der eine Diskette emuliert. SPEAK: stellt den Sprach-Handler dar, der eine Sprachausgabe des Amiga ermöglicht. Sogar die Darstellungsform von Fenstern am Bildschirm wird durch Handler gesteuert (CON: und NEWCON:).

In diesem Abschnitt werden teilweise Informationen wiederholt, die an anderer Stelle dieses Handbuchs ebenfalls vorhanden sind, wie z. B. die Eigenschaften des Handlers NEWCON:, der im Abschnitt über die SHELL beschrieben ist. Die vollständigen Erklärungen zu den neuen Handlern sind jedoch nachstehend aufgeführt.

3.4.1 Aux-Handler

Der Handler AUX: ermöglicht serielle Ein- und Ausgabe ohne Pufferung. Er wird hauptsächlich als Konsolen-Handler verwendet, der anstelle des Amiga-Bildschirms und der Amiga-Tastatur den seriellen Anschluß benutzt.

Der Eintrag in MountList lautet:

```
AUX:    Handler= L:Aux-Handler
        Stacksize = 1000
        Priority = 5
#
```

In der Datei MountList ist bereits ein Beispieleintrag enthalten.

Sie können den Aux-Handler verwenden, um eine andere Datenstation mit Ihrem Computer zu benutzen. Zum Beispiel:

```
1>  MOUNT  AUX:
1>  NEWCLI  AUX:
```

ACHTUNG: Der Aux-Handler erlaubt Ihnen, mehrfach den Befehl NEWSHELL oder NEWCLI zu verwenden. Die Tastaturdaten von der Aux-Einheit werden jedoch an jeden der aufgerufenen Befehlsprozessoren übergeben. Soll eine von zwei an AUX: zugewiesene SHELLs geschlossen werden, ist einzugeben:

```
1> E N D C L I <Leerschritt> <RETURN>
```

Nach jedem Buchstaben des Wortes und nach dem "I" am Ende muß ein Leerschritt eingegeben werden!

Soll eine Datei über den seriellen Anschluß kopiert werden, ist einzugeben:

```
1> COPY AUX: to RAM:testdatei
```

3.4.2 Newcon-Handler

Der Handler NEWCON: wirkt als Ersatz für den Handler CON: und wird durch den Namen NEWCON: angesprochen. Ebenso wie die anderen neuen Handler muß er vor der Verwendung durch den Befehl MOUNT angemeldet werden. Dies erfolgt im allgemeinen durch einen Befehl in der Startdatei. (Die Workbench-Version 1.3 enthält einen solchen Befehl in der Startup-Sequence.)

Der Eintrag in MountList lautet:

```
NEWCON:    Handler= L:Newcon-Handler
           Priority = 5
           Stacksize = 1000
#
```

Ein NEWCON:-Fenster bietet dem Benutzer erheblich mehr Flexibitität bei der Befehlseingabe. Im Gegensatz zu CON:, bei dem bei Schreibfehlern alle Zeichen bis zum Fehler gelöscht werden müssen, erlaubt ein NEWCON:-Fenster das Ändern der eingegebenen Befehlszeile(n). Ein Beispiel für ein aktives NEWCON:-Fenster ist die SHELL. Nachstehend eine Liste der Fähigkeiten von NEWCON:-Fenstern:

1) Bearbeitung von Befehlszeilen

   Die Pfeiltasten "Cursor nach links" und "Cursor nach rechts" können verwendet werden, um den Cursor in der Befehlszeile zu bewegen. Die Taste BACKSPACE löscht das Zeichen links vom Cursor, die Taste DELETE das Zeichen an der Cursorposition.

Einige weitere Befehle für die Bearbeitung lauten:

Control-K	Löscht den gesamten Text von der Cursorposition bis zum Zeilenende.
Control-U	Löscht den gesamten Text von der Cursorposition bis zum Zeilenanfang.
Control-X	Löscht die gesamte Zeile.
Control-W	Bewegt den Cursor zur nächsten Tabulatorposition.

Die Tastenkombination Shift-"Cursor nach links" (oder Control-A) bewegt den Cursor zum Anfang der Zeile. Durch Shift-"Cursor nach rechts" wird er zum Zeilenende bewegt.

2) Befehlspuffer

NEWCON: verfügt über einen zirkularen "Gedächtnisspeicher" mit 2 KByte Kapazität. Zuvor eingegebene Befehle können durch Verwendung der Taste "Cursor aufwärts" wieder abgerufen werden. (Die Taste "Cursor abwärts" bewegt den Cursor in Vorwärtsrichtung durch den Befehlspuffer. Dies ist nützlich, wenn Sie weiter als bis zum gesuchten Befehl zurückgeblättert haben.)

Die Tastenkombination Shift-"Cursor aufwärts" (oder Control-R) bewirkt, daß in Rückwärtsrichtung nach dem letzten Befehl gesucht wird, der einer als Muster teilweise eingegebenen Zeichenfolge entspricht. Die Tastenkombination Shift-"Cursor abwärts" bewegt den Cursor zum Ende des Befehlspuffers.

3) Steuerzeichen

Steuerzeichen werden unter NEWCON: durch inverse Zeichen (Steuerzeichencode + 64) dargestellt. Die Zeichen werden als Steuerzeichen gelesen und können wie bisher in Dateien verwendet werden.

3.4.3 Pipe-Handler

PIPE: ist ein von Matt Dillon geschriebener AmigaDOS-Handler.

Der Pipe-Handler ist ein Ein-/Ausgabeprogramm, das die Ein-/Ausgabe-Kommunikation zwischen Programmen ermöglicht. Die grundlegende Funktion ist die Erstellung eines Kommunikationskanals zwischen Prozessen. Wenn dieser Kanal als Ziel für einen Schreibvorgang angegeben wird, werden bis zu 4 KByte

Daten zwischengespeichert, bevor der Schreibvorgang blockiert wird. Nach dem Schreiben zu PIPE: können die Daten sofort durch einen anderen Prozeß gelesen werden.

Ebenso wie die anderen Handler muß er durch den Befehl MOUNT angemeldet werden. Der Eintrag für PIPE: in der Datei MountList lautet wie folgt:

```
PIPE:    Handler = L:Pipe-Handler
         Stacksize = 6000
         Priority = 5
         GlobVec = -1
#
```

PIPE: kann auch von anderen Programmen, wie z. B. Textverarbeitungsprogrammen (als Dateiname für Datenspeicherung) oder Terminalprogrammen (als Dateiname für einen Protokollierungs-Puffer) verwendet werden. PIPE: verwendet einen internen Puffer mit 4 KByte Kapazität für jeden Datenkanal. Die optimale Situation ist jedoch gegeben, wenn die Lese- und Schreibvorgänge gleichzeitig ablaufen. In diesem Fall werden die Daten direkt von einem Prozeß zu einem anderen kopiert, ohne den Umweg über den internen Puffer. Dies ist ein echter Übertragungskanal, weshalb der Quellen- und der Zielprozeß zwei getrennte Prozesse sein müssen, damit sie sich nicht gegenseitig sperren.

Der Puffer ist transparent. Das bedeutet, daß geschriebene Daten, ohne Rücksicht auf eine Mindestmenge, unmittelbar zum Lesen durch einen anderen Prozeß zur Verfügung stehen.
Das Gerät PIPE: (abgeleitet von "Pipeline") kann nützlich sein, wenn Sie mit zwei Anwendungsprogrammen arbeiten und riesige Datenmengen von einem (schreiben) zum anderen (lesen) übertragen wollen, ohne eine temporäre Datei in RAM: oder auf einer Diskette anzulegen. Vorausgesetzt, das Anwendungsprogramm versucht nicht die Anweisung Seek() durchzuführen, kann einfach "PIPE:name" angegeben werden, was für das Anwendungsprogramm wie eine gewöhnliche Datei wirkt. Für jede PIPE:-Datenverbindung muß also ein eigener Verbindungsname angegeben werden zur eindeutigen Identifizierung.

Für Terminalprogramme, die keinen asynchronen Schreibmodus verwenden, können die Unterbrechungen beim Schreibvorgang durch PIPE: abgefangen werden. Die Daten müssen dann durch einen weiteren Befehl COPY von PIPE: zu einer Datei kopiert werden.

Informationen können auch von einer PIPE:-Anwendung zu einer anderen kopiert werden. Zum Beispiel:

```
CLI-Fenster 1:   COPY riesendatei PIPE:a
CLI-Fenster 2:   COPY PIPE:a PIPE:b
CLI-Fenster 3:   COPY PIPE:b PIPE:c
CLI-Fenster 4:   COPY PIPE:c PIPE:d
CLI-Fenster 5:   COPY PIPE:d PIPE:e
CLI-Fenster 6:   wortzählung PIPE:e  (oder ähnliches)
```

3.4.4 Speak-Handler

Der Speak-Handler ermöglicht dem Amiga Sprachausgabe. Durch SPEAK: können Sie sich im wörtlichen Sinn den Inhalt einer Datei vom Amiga "vorlesen" lassen.

Ebenso wie die anderen Handler muß SPEAK: vor der Verwendung durch den Befehl MOUNT angemeldet werden. Der Eintrag in die Datei MountList lautet:

```
SPEAK: Handler = L:Speak-handler
       Stacksize = 4000
       Priority = 5
       GlobVec = -1
#
```

Zusätzlich zum Eintrag in MountList erfordert SPEAK: auch das Gerät NARRATOR.DEVICE und die Bibliothek TRANSLATOR.LIBRARY. Sie müssen sich in den Verzeichnissen DEVS: bzw. LIBS: befinden. (Diese Dateien sind bei der Workbench-Version 1.3 in den korrekten Verzeichnissen enthalten.)

Das Format für den Aufruf von SPEAK: lautet:

```
SPEAK:OPT/K
```

Nach dem Schlüsselwort OPT können die folgenden Optionen angegeben werden. Die einzelnen Optionen müssen durch einen Schrägstrich (/) getrennt sein und es darf kein Leerschritt zwischen dem Doppelpunkt und dem Wort OPT vorhanden sein.

p###	Stimmlage (Wertebereich für ###: 65 bis 320)
s###	Geschwindigkeit (Wertebereich für ###: 30 bis 400)
m	Männliche Stimme
f	Weibliche Stimme
r	Roboterstimme
n	Natürliche Stimme
o0	Diese Optionen im Eingabestrom nicht erlauben.
o1	Diese Optionen im Eingabestrom erlauben.
a0	Den Modus Direkt-Phonem ausschalten.
a1	Den Modus Direkt-Phonem einschalten (TRANSLATOR.LIBRARY nicht verwenden.)
d0	Sätze nur bei Zeichensetzung beenden.
d1	Sätze bei Zeichensetzung, RETURN und LINEFEED beenden.

SPEAK: kann auch von anderen Programmen, wie z. B. Textverarbeitungsprogrammen (als Dateiname für Datenspeicherung) oder Terminalprogrammen (als Dateiname für einen Datenerfassungs-Puffer) verwendet werden, um eine gesprochene Ausgabe zu erhalten.

Wenn Sie z. B. den Inhalt der Datei mit der Startsequenz abhören wollen, geben Sie ein:

```
1> MOUNT SPEAK:
1> COPY s:startup-sequence to SPEAK:OPT/f/s160
```

Der Inhalt der Startsequenz wird mit weiblicher Stimme und mäßiger Geschwindigkeit gelesen.

3.5 SHELL-SEG

Obwohl sich SHELL-SEG zusammen mit den Amiga-Handlern im Verzeichnis L: befindet, ist es kein Handler. Es ist das Programm, das die neue SHELL steuert. SHELL-SEG wird nicht direkt ausgeführt. Statt dessen wird der Befehl RESIDENT verwendet, um es unter dem Namen CLI aufzurufen. Sobald es gestartet ist, wird für jede neue SHELL, die entweder durch den Befehl NEWSHELL oder das SHELL-Piktogramm aktiviert wird, ein SHELL-Befehlsprozessor anstatt eines normalen Befehlsprozessors CLI aufgerufen. Normalerweise wird SHELL-SEG durch die Startsequenz aufgerufen.

Die SHELL bietet viele zusätzliche Merkmale, die der Standardbefehlsprozessor CLI nicht enthält. Die meisten dieser Merkmale sind im Abschnitt SHELL in Kapitel 1 beschrieben, wie z. B. residente Befehle, Makros, die Möglichkeit, die Eingabeaufforderung zu ändern und die Verwendung von Befehlsdateien.

3.6 Befehlsdateien (Scripts)

Das Verzeichnis S: der Workbench-Diskette enthält einige Befehlsdateien. Sie können zusammen mit anderen Befehlen verwendet werden, um den Ablauf Ihrer Programme zu beschleunigen. Diese Dateien sind nachstehend aufgelistet.

SPAT — Verwendet den Befehl LIST, um Befehle mit genau einem Argument mit der Möglichkeit des Mustervergleichs auszustatten. Dies kann mit der Makro-Funktion von SHELL erfolgen. Sollen z. B. die Inhalte aller Befehlsdateien, die mit einem "s" beginnen und sich im Verzeichnis S: der Workbench-Diskette befinden aufgelistet werden, kann eingegeben werden:

```
1> SPAT MORE s:s#?
```

Eine Befehlsdatei, ähnlich der nachstehenden, wird erstellt:

```
more "s:SPAT"
more "s:Startup-Sequenz.hd"
more "s:Shell-Startup"
more "s:Startup-Sequenz"
more "s:StartupII"
```

SPAT würde dann diese Befehlsdatei ausführen und den Inhalt der Dateien nacheinander anzeigen.

DPAT — Verwendet den Befehl LIST, um Befehle mit zwei Argumenten mit der Möglichkeit des Mustervergleichs auszustatten. Dies kann mit der Makro-Funktion von SHELL erfolgen. Beim folgenden Beispiel wird vorausgesetzt, daß eine Serie von Dateien mit den Namen Kapitel1, Kapitel2, Kapitel3, etc. im Wurzelverzeichnis der Diskette in df0: vorhanden ist. Sollen diese Dateien in ein neues Verzeichnis mit dem Namen "Buch" eingeordnet werden, lautet die Befehlsfolge:

```
1> MAKEDIR df0:BUCH (erstellt das neue Verzeichnis)
1> DPAT RENAME df0:Kap#? df0:buch/
```

Eine Befehlsdatei wird erstellt, um alle Dateien umzubenennen, deren Name mit "Kap" beginnt. DPAT führt dann diese Datei aus und versieht jede "Kapitel-Datei" mit der Verzeichnisbenennung BUCH.

PCD — Ähnlich zum AmigaDOS-Befehl CD, "erinnert" sich jedoch an das Ausgangsverzeichnis. Zum Beispiel:

```
1> PCD RAM:
1> PCD
```

Es erfolgt die Rückkehr zum vorherigen Verzeichnis.

3.7 Bibliotheken (Libraries)

Im Verzeichnis LIBS: der Workbench-Diskette der Version 1.3 befinden sich neue IEEE-Bibliotheken: mathieeedoubtrans.library und mathieeedoubbas.library. Diese neuen Bibliotheken können einen Koprozessor 68881, falls vorhanden, nützen. Sie können auch mit einem peripheren Mathematik-Baustein arbeiten; der Baustein muß jedoch mit der entsprechenden Software geliefert werden, die das System über das Vorhandensein des Bausteins informiert und beim Umschalten zwischen Tasks das Speichern und Laden des Baustein-Status übernimmt.

4. Druckertreiber

Die Version 1.3 unterstützt mehr Druckertreiber als die Version 1.2. Es wurden jedoch alle Treiber mit Ausnahme von Generic auf der Extras-Diskette untergebracht. Dadurch steht mehr Raum auf der Workbench-Diskette zur Verfügung, wobei der Amiga-Benutzer gleichzeitig unter mehr Druckertreibern wählen kann.

Ein Druckertreiber arbeitet wie ein Übersetzer für den Computer. Der Amiga verwendet eine einheitliche, geräteunabhängige Art der Informationscodierung (ANSI-Sequenzen), die Drucker würden jedoch diesen Code mit unterschiedlichen Ergebnissen umsetzen. Der Druckertreiber übernimmt die Informationen vom Amiga und arbeitet sie so um, daß der angeschlossene Drucker sie korrekt interpretiert.

Die Druckertreiber befinden sich im Verzeichnis DEVS/PRINTERS auf der Extras-Diskette. Soll ein Treiber zur Workbench-Diskette übertragen werden, kann entweder das Hilfsprogramm InstallPrinter (in Kapitel 1 erklärt) oder der AmigaDOS-Befehl COPY (wie nachstehend angegeben) verwendet werden.

```
COPY Extras1.3D:devs/printers/<Treibername> to  devs:printers
```

Nach dem Kopieren der Datei erscheint der Treiber in der Liste der unterstützten Treiber, die im Fenster ChangePrinter des Voreinstellerprogramms Preferences angezeigt wird.

Die Version 1.3 enthält die folgenden Druckertreiber:

Alphacom AlphaPro 101
Brother HR-15XL
CalComp ColorMaster
CalComp ColorMaster2
Canon PJ-1080A
CBM MPS1000
Diablo 630
Diablo Advantage D25
Diablo C-150
Epson Q
Epson X
EpsonXOld
Generic*
Howtek Pixelmaster
HP DeskJet
HP LaserJet (inkl. HP-LaserJet-Plus- und HP-LaserJet-II-Kompatible)
HP PaintJet
HP ThinkJet
Imagewriter II
NEC Pinwriter
Okidata 92
Okidata 293I
Okimate 20
Quadram QuadJet
Qume LetterPro 20
Toshiba P351C
Toshiba P351SX
Xerox 4020

* Auf der Workbench-Diskette enthalten.

Spezifikationen zu jedem dieser Treiber sind später in diesem Kapitel zu finden. Es enthält auch Informationen über die Druckertreiber für den Seiko 5300 und 5300a und den Tektronix 4693D und 4696. Obwohl diese Treiber nicht auf der Diskette vorhanden sind, sind sie kompatibel mit dem Amiga und durch den Hersteller erhältlich, als Public-Domain-Software oder über BIX, den elektronischen Informationsaustausch des Byte-Magazins in USA.

Vor der Auflistung der einzelnen Treiber sind nachstehend allgemeine Informationen, den Grafikdruck betreffend, aufgeführt. Bevor Sie mit diesem Abschnitt des Handbuchs fortfahren, sollten Sie den Abschnitt über das Voreinstellerprogramm Preferences in Kapitel 1 lesen. Er enthält Einzelheiten über die Farbkorrektur, das Glätten, mehrere Druckdichten und das Ändern der Größe des Ausdrucks.

4.1 Grafikdruck

Der Amiga ist weltweit für seine Grafikfähigkeiten bekannt. Wenn Sie jedoch nicht in der Lage sind, die Grafiken vom Bildschirm auf Papier zu übertragen, ist die Anwendung eingeschränkt. Ob es sich nun um eine Firmenzeitung, technische Zeichnungen oder einfach nur um eine selbst hergestellte Grußkarte handelt - wenn die Bildschirmanzeige nicht gedruckt werden kann, können Sie Ihre Arbeit nicht weitergeben. Der Amiga ist in der Lage, verschiedene Drucker zu unterstützen - vom einfachen Typenraddrucker, der nur Texte in Schwarz/Weiß drucken kann, bis zum hochentwickelten Farb-Thermodrucker.

4.1.1 Geschwindigkeit

Die Druckertreiber der Version 1.3 arbeiten ca. fünf- bis dreißigmal schneller als die der Version 1.2. Die Geschwindigkeitserhöhung ist abhängig von der Art des verwendeten Druckers, der Anzahl der Druckfarben (schwarz/weiß, Grauwerte oder Farbe), dem Umfang der weißen Flächen innerhalb der Abbildung, der Art der Darstellung (verikal oder horizontal) und dem eingesetzten Druckertreiber. Die Druckgeschwindigkeit hängt nun mehr vom Drucker oder der benutzten Software und weniger vom eigentlichen Computer ab.

4.1.2 Farbdruck

HAM-Bilder - Ein HAM-Bild (hold-and-modify) ist eine Abbildung, in der 32 bis zu allen 4.096 Farben des kompletten Amiga-Farbenbereichs von angezeigt werden. Bisher konnte ein HAM-Bild nur mit der X-Position = 0 (entspricht der oberen linken Ecke des Papiers) gedruckt werden. Nun kann der Druck an jeder gültigen X-Position gestartet werden. Es ist auch möglich, invertierte HAM-Bilder zu drucken.

Nachstehend einige Hinweise, die beim Ausdruck von Grafikbildschirmen berücksichtigt werden sollten:

Für höhere Qualität von Ausdrucken des Grafikbildschirms:

- Bei den meisten Druckern wird auf durch Reibung transportiertem Papier der Grafikbildschirm hochwertiger gedruckt als auf durch einen Traktor transportiertem Papier. Es ergeben sich weniger horizontale Streifen.

- Druckdichten, bei denen mehr als ein Durchlauf erfolgt, sollten nur für Schwarz/Weiß-Ausdrucke des Grafikbildschirms verwendet werden. (Die Dichte wird durch die Einstellung von Density im Voreinstellerprogramm gesteuert. Die bei Ihrem Drucker möglichen Einstellungen entnehmen Sie bitte der Liste der Druckertreiber später in diesem Kapitel.) Wird eine Dichte, die mehrfachen Durchlauf verlangt, für den Ausdruck eines Grafikbildschirms mit Grauwerten oder Farbschattierungen verwendet, kann das Druckergebnis unscharf oder dunkel sein. Farbgrafikausdrucke mit mehrfachen Durchläufen verschmutzen auch das Farbband des Druckers (d. h. die Farbe Gelb wird durch andere Farben des Bandes verunreinigt).

Für schnelleren Druck von Grafikbildschirmen:

- Je geringer die Dichte, um so schneller der Druck.
- Ausdrucke im Horizontalformat werden erheblich schneller gedruckt als solche im Vertikalformat.
- Wenn Sie den Ausdruck einer Abbildung mit einer Bitebene (zwei Farben) erstellen wollen, sollten Sie im Voreinstellerprogramm den Modus "Black and White" (Schwarz/Weiß) wählen. Er arbeitet bedeutend schneller als die Einstellungen "Grey Scale" (Grauwerte) oder "Color" (Farbe).
- Ist die Option "Smoothing" (Glätten) aktiv, verdoppelt sich die Druckzeit. Verwenden Sie diese Option nur für den endgültigen Ausdruck.

- Wählen Sie die Art der Schattierungsdarstellung (Dithering) immer dem gewünschten Ergebnis entsprechend. Bei der Einstellung "F-S" verdoppelt sich die Druckzeit, während sich die Einstellungen "Ordered" (geordnet) und "Halftone" (Halbton) nicht negativ auf die Geschwindigkeit auswirken.
- Soll ein Grafikbildschirm mit niedriger Auflösung (320 x 256 oder 512 Bildpunkte) mit mehr als 16 Farben (4 Bitebenen) oder mit höher Auflösung (640 x 256 oder 512 Bildpunkte) und mehr als 4 Farben (2 Bitebenen) gedruckt werden, kann der Ausdruck wie folgt beschleunigt werden: Bewegen Sie den Bildschirm in den Hintergrund, SOBALD DER DRUCK BEGONNEN HAT. Dies wird durch die gleichzeitige Betätigung der linken Amiga-Taste und der Taste N erreicht.

4.2 Unterstützte Drucker

Nachstehend ist eine Liste der Druckertreiber aufgeführt, die für die Verwendung mit Amiga-Computern verfügbar sind. Viele dieser Treiber befinden sich auf der Extras-Diskette. Die wenigen Treiber, die dort nicht vorhanden sind, können über die oben genannten Wege bezogen werden.

Sie werden feststellen, daß viele der Treiber unterschiedliche Druckdichten unterstützen. Die Dichte bezieht sich auf die Anzahl der Punkte pro Inch, die bei der Erstellung des Ausdrucks verwendet wird. Je höher die Anzahl der Punkte ist, um so kleiner werden sie und um so klarer das gedruckte Bild. Jedoch gilt auch: je höher die Anzahl der Punkte, um so langsamer der Druck. Deshalb muß der Benutzer, sofern unterschiedliche Druckdichten zur Wahl stehen, selbst entscheiden, ob schneller oder mit höherer Qualität gedruckt werden soll.

Sofern anwendbar, werden für viele Drucker Diagramme für die Standard-Schalterstellung angegeben.

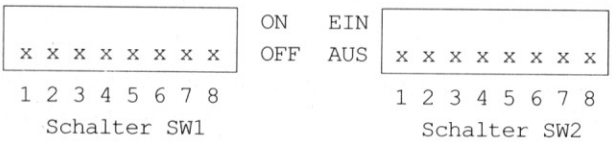
HINWEIS: In den nachstehenden Beschreibungen der Druckertreiber wird die Anzahl der Punkte pro Inch (DPI, dots per inch), falls veränderlich, in Tabellenform angegeben. Dabei gilt: XDPI = Anzahl der in horizontaler Richtung gedruckten Punkte pro Inch, YDPI = Anzahl der in vertikaler Richtung gedruckten Punkte pro Inch, XYDPI = Anzahl der gedruckten Punkte pro Inch^2. Die dort angesprochenen Hinweise 1 bis 3 finden Sie am Schluß dieser Druckerliste.

Alphacom__AlphaPro__101

- Typenraddrucker, nur Druck von Texten.
- Schalterstellung:
  MODUS A: ON/EIN (bidirektionaler Druck)
  MODUS B: OFF/AUS (kein Zeilenvorschub nach Zeilenrücklauf)
  Getestete Schnittstellenplatine: 1011PC (IBM PC)

Brother__HR-15XL

- Typenraddrucker, nur Druck von Texten.
- Schalterstellungen:



Calcomp__ColorMaster

- Thermodrucker, S/W- und Farbdruck, druckt Text und Grafik.
- Nur eine Dichte wird unterstützt - 203 x 200 DPI. Die Dichte 1 wählen.
- Dies ist ein Druckertreiber für zwei Drucker. Setzen Sie die Einstellung von "Paper Size" bei Verwendung des Druckers ColorMaster zu "Narrow Tractor" und bei Verwendung des Druckers ColorView-5912 (für die Papiergröße von 11 x 17 Inch) zu "Wide Tractor".
- Es sind keine DIP-Schalter vorhanden.

Calcomp__ColorMaster2

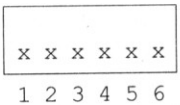
- Thermodrucker, S/W- und Farbdruck, druckt Text und Grafik.
- Nur eine Dichte wird unterstützt - 203 x 200 DPI. Die Dichte 1 wählen.

- Dies ist ein Druckertreiber für zwei Drucker. Setzen Sie die Einstellung von "Paper Size" bei Verwendung des Druckers ColorMaster zu "Narrow Tractor" und bei Verwendung des Druckers ColorView-5912 (für die Papiergröße von 11 x 17 Inch) zu "Wide Tractor".
- Dieser Treiber entspricht genau dem Treiber Calcomp__ColorMaster, MIT DER AUSNAHME, daß dieser beim Drucken von Farbgrafiken etwa mit doppelter Geschwindigkeit arbeitet und einen großen Speicherbereich erfordert (bis zu 1.272.003 Byte für einen Farbdruck in voller Größe bei 8 x 10 Inch [1.600 x 2.000 Punkte]). Beim typischen Druck in voller Größe umfassen Ausdrucke von Grafikbildschirmen 1.600 x 1.149 Punkte und erfordern 730.767 Byte.
  Beim ColorView-5912 sind bis zu 2.572.803 Byte für einen Farbdruck in voller Größe bei 10 x 16 Inch (2.048 x 3.200 Punkte) erforderlich. Beim typischen Druck in voller Größe umfassen Ausdrucke von Grafikbildschirmen 2.048 x 2.155 Punkte und erfordern 1.732.623 Byte.
  Wird ein Ausdruck ohne Farbe erstellt, ist etwa ein Drittel des Speichers erforderlich, der für einen vergleichbaren Farbdruck benötigt wird.
- Es sind keine DIP-Schalter vorhanden.

Canon__PJ-1080A

- Tintenstrahldrucker S/W- und Farbdruck, druckt Text und Grafik.
- Nur eine Dichte wird unterstützt - 83 x 84 DPI. Die Dichte 1 wählen.
- Schalterstellungen:

Canon PJ-1080A



ON EIN
OFF AUS

CBM_MPS1000 (Auch für Drucker, die mit dem IBM 5152 kompatibel sind)

- Matrixdrucker, ohne Farbdruck, druckt Text und Grafik
-

Dichte	XDPI	YDPI	XYDPI	Kommentar
1	120	72	8.640	
2	120	144	17.280	Zwei Durchläufe
3	240	72	17.280	Siehe Hinweis 2
4	120	216	25.920	Drei Durchläufe; siehe Hinweis 2
5	240	144	34.560	Zwei Durchläufe; siehe Hinweis 2

6	240	216	51.840	Drei Durchläufe; siehe Hinweis 2
7				Wie bei Dichte 6

- Schalterstellungen:

CBM MPS1000

```
x
  x x x x x x x      ON   EIN
                     OFF  AUS
1 2 3 4 5 6 7 8
```

Canon BJ-130 (mit Steuerkapsel 48/XL - kompatibel mit IBM Proprinter)

```
x x x x x x x x x x      ON   EIN      x x x x x x x x x x
                         OFF  AUS
1 2 3 4 5 6 7 8 9 10                    1 2 3 4 5 6 7 8 9 10
    Schalter SW1                            Schalter SW2
```

HINWEIS: Wenn Sie einen Drucker Commodore MPS1250 besitzen, siehe Informationen über EpsonX. Diese beiden Drucker sind kompatibel.

Diablo__630

- Typenraddrucker, nur Druck von Texten.
- Schalterstellungen:

```
                         ON   EIN
x x x x x x x x          OFF  AUS      x x x x x x x x
1 2 3 4 5 6 7 8                        1 2 3 4 5 6 7 8
  Schalter SW1                           Schalter SW2
```

Diablo__Advantage__D25

- Typenraddrucker, druckt nur Texte
- Schalterstellungen:
  Die Parallelschnittstelle wird empfohlen.

	AUS OFF	EIN ON
1	x	
2	x	
3	x	
4	x	
5	x	
6	x	
7	x	
8	x	

LINKS

	AUS OFF	EIN ON
1	x	
2	x	
3	x	
4	x	
5	x	
6	x	
7	x	
8	x	

RECHTS

Der linke Schalter ist für das XON/XOFF-Protokoll der seriellen Schnittstelle und die Unterdrückung des Zeilenvorschubs nach einem Zeilenrücklauf eingestellt.

Diablo__C-150

- Tintenstrahldrucker S/W- und Farbdruck, druckt Text und Grafik.
- Zeilenvorschub entspricht immer 4 Punkten (Einschränkung des Druckers).
- Nur eine Dichte wird unterstützt - 120 x 120 DPI. Die Dichte 1 wählen.
- Wird die Einstellung von "Paper Size" im Voreinstellerprogramm mit "Wide Tractor" gewählt, beträgt die maiximale Druckbreite 8,5 Inch (bei breiter Papierrolle).
- Schalterstellungen:

```
    x x x          ON   EIN
x x       x x x    OFF  AUS
1 2 3 4 5 6 7 8
 Schalter SW1
```

EpsonQ (Drucker mit 24 Nadeln, kompatibel mit der Epson-Serie "Q")

- Matrixdrucker S/W- und Farbdruck, druckt Text und Grafik.
- Treiber für alle Drucker, die kompatibel mit EpsonQ (LQ1500, LQ2500, etc.) sind.
-

Dichte	XDPI	YDPI	XYDPI	Kommentar
1	90	180	16.200	
2	120	180	21.600	
3	180	180	32.400	

Okimate-20

- Thermodrucker, S/W- und Farbdruck, druckt Text und Grafik.
- Zeilenvorschub ist immer eine gerade Anzahl von Punkten (d. h. wenn drei Punkte gedruckt wurden, Vorschub um 4 Punkte).
- Nur eine Dichte wird unterstützt - 120 x 144 DPI. Die Dichte 1 wählen.
- Schalterstellungen:
  Einbausatz "Parallel Plug'n Print"

```
          x        ON   EIN
x x x x   x        OFF  AUS
1 2 3 4 5 6
```

HINWEIS: Der Schalter 5 steuert bei einigen Modellen den weißen Bereich zwischen den Zeilen eines Grafikdrucks.

Einbausatz "Serial Plug'n Print"

```
x x       x        ON   EIN        x
    x x x          OFF  AUS    x x x x   x
1 2 3 4 5 6                    1 2 3 4 5 6
Schalter SW1                   Schalter SW2
```

HINWEIS: Die Einstellungen des Schalters 1 legen eine Baudrate von 9.600, das XON/XOFF-Protokoll, eine Wortlänge von 8 Bit und keine Paritätsprüfung fest. Der DIP 5 des Schalters 2 steuert bei einigen Modellen den weißen Bereich zwischen den Zeilen eines Grafikdrucks.

Quadram__QuadJet

- Tintenstrahldrucker, S/W- und Farbdruck, druckt Text und Grafik.
- Nur eine Dichte wird unterstützt - 83 x 84 DPI. Die Dichte 1 wählen.
- Es sind keine DIP-Schalter vorhanden.

Qume__LetterPro__20

- Typenraddrucker, nur für den Druck von Texten.
- Schalterstellungen:

```
                x
x x x x x x    x
1 2 3 4 5 6 7 8
 Schalter SWA
```

```
ON   EIN
OFF  AUS
```

```
  x x
x     x x
1 2 3 4 5
Schalter SWB
```

Seiko__5300

DIESER TREIBER IST NICHT AUF DER WORKBENCH- ODER DER EXTRAS-DISKETTE VORHANDEN. ER IST VON SEIKO ODER ÜBER BIX (S. O.) ERHÄLTLICH.

- Thermodrucker, S/W- und Farbdruck, druckt nur Grafik.
-

Dichte	XDPI	YDPI	XYDPI	Kommentar
1	152	152	23.104	Treiber für Drucker CH-5301
2	203	203	41.209	Treiber für Drucker CH-5312
3	240	240	57.600	Treiber für Drucker CH-5303
4	Wie bei Dichte 3			
5	Wie bei Dichte 3			
6	Wie bei Dichte 3			
7	Wie bei Dichte 3			

- Keine DIP-Schalter vorhanden

Seiko__5300a

DIESER TREIBER IST NICHT AUF DER WORKBENCH- ODER DER EXTRAS-DISKETTE VORHANDEN. ER IST VON SEIKO ODER ÜBER BIX (S. O.) ERHÄLTLICH.

- Thermodrucker, S/W- und Farbdruck, druckt nur Grafik.
-

Dichte	XDPI	YDPI	XYDPI	Kommentar
1	152	152	23.104	Treiber für Drucker CH-5301
2	203	203	41.209	Treiber für Drucker CH-5312
3	240	240	57.600	Treiber für Drucker CH-5303
4	Wie bei Dichte 3			
5	Wie bei Dichte 3			
6	Wie bei Dichte 3			
7	Wie bei Dichte 3			

- Dieser Treiber entspricht genau dem Treiber Seiko__5300, MIT DER AUSNAHME, daß dieser beim Drucken von Farbgrafiken etwa mit doppelter Geschwindigkeit arbeitet und einen großen Speicherbereich erfordert (bis zu

1.564.569 Byte für einen Farbdruck in voller Größe bei 8 x 10 Inch [1.927 x 2.173 Punkte]). Beim typischen Druck in voller Größe umfassen Ausdrucke von Grafikbildschirmen 1.927 x 1.248 Punkte und erfordern 898.569 Byte. Ein Ausdruck ohne Farben erfordert etwa 1/3 des Speichers, der für einen vergleichbaren Farbdruck benötigt wird.

- Es sind keine DIP-Schalter vorhanden.

Tektronix__4693D

DIESER TREIBER IST NICHT AUF DER WORKBENCH- ODER DER EXTRAS-DISKETTE VORHANDEN. ER IST VON TEKTRONIX ODER ÜBER BIX (S. O.) ERHÄLTLICH.

- Thermodrucker, S/W- und Farbdruck, druckt nur Grafik.
- Nur eine Dichte wird unterstützt - 300 x 300 DPI. Die Dichte 1 wählen.
- Bedingt durch die Art, in der der Drucker eine Abbildung erstellt, beeinflußt keine der Optionen des Drucker-Voreinstellers den Ausdruck, mit folgenden Ausnahmen:
  a)Aspect - Horizontal, Vertical
  b)Shade - Black and White, Grey Scale, Color
  Deshalb können nur Abbildungen in voller Größe gedruckt werden.

Tektronix__4696

DIESER TREIBER IST NICHT AUF DER WORKBENCH- ODER DER EXTRAS-DISKETTE VORHANDEN. ER IST VON TEKTRONIX ODER ÜBER BIX (S. O.) ERHÄLTLICH.

- Tintenstrahldrucker, S/W- und Farbdruck, druckt Text und Grafik.
- Der Zeilenvorschub umfaßt immer 4 Punkte (Beschränkung des Druckers).

- Der Drucker unterstützt drei Dichten:
  121 x 120 DPI Die Dichte 1 wählen. Die Ausgabe aller Farben erfolgt in einem Durchlauf.
  242 x 120 DPI S/W-Druck. Die Dichte 2 wählen. Es werden zwei Durchläufe für Schwarz durchgeführt.
  242 x 120 DPI Farbdruck. Die Dichte 3 wählen. Es werden zwei Durchläufe für alle Farben durchgeführt.
  Die Wahl der Dichte 2 oder höher bewirkt nicht wirklich eine Auflösung von 242 DPI, da der Drucker nur 121 DPI unterstützt. Statt dessen wird der Drucker angewiesen, einen zweiten Durchlauf auszuführen. Beim zweiten Durchlauf

Datei Startup-Sequence

Öffnen einer Anfangs-SHELL
Festlegen zusätzlicher Pfade und logischer Namen
Anmelden von Festplatten-Partitionen
Anlegen einer resetfesten RAM-Disk (RAD:)
Übergabe der Steuerung an ein RAD:, die Workbench-Diskette oder eine Festplatte.

Datei StartupII

RAM: umbenennen; für Programme, bei denen die Angabe eines Leerschritts in einem Datenträgernamen nicht erlaubt ist (Standard-Datenträgername ist "RAM DISK")
Weitere Befehle resident laden
Umgebungsvariablen festlegen wie z. B. ENV:Editor (vom Hilfsprogramm MORE verwendet)

Datei Shell-Startup

Eigene Befehlsmakros in der Datei S:SHELL-STARTUP festlegen

In diesem Kapitel sind einige Beispiele für die Veränderung dieser Dateien mit Startsequenzen aufgeführt, außerdem Anleitungen für den Neustart ausgehend von der resetfesten RAM-Disk. Bevor Sie versuchen, eine der Start-Befehlsdateien zu verändern, sollten Sie unbedingt eine Sicherungskopie der Workbench-Diskette erstellen. Sie sollten auch beachten, daß das System im allgemeinen im Verzeichnis S: nach Befehlsdateien sucht. Wenn Sie während oder nach dem Start S: einem anderen Gerät zuweisen (wie z. B. einer Festplatte oder der resetfesten RAM-Disk), müssen die veränderten Befehlsdateien ebenfalls in diesem Verzeichnis plaziert werden. Wir empfehlen Ihnen, Sicherungskopien aller veränderten Startdateien und MountList-Dateien auf einer Diskette aufzubewahren.

Durch die folgenden Beispiele werden einige einfache Änderungen beschrieben, die bezüglich der Start-Befehlsdateien vorgenommen werden können.

5.1 Öffnen einer Anfangs-SHELL

Fügen Sie die folgenden Zeilen vor der Zeile 22 "endcli >NIL:" am Ende der Datei Startup-Sequence ein:

```
cd SYS: ; Falls gewünscht, können Sie auch cd RAM: eingeben.
NewShell newcon:0/0/640/200/AmigaShell
```

5.2 Festlegen zusätzlicher Pfade und logischer Namen

Pfade und logische Zuweisungen zu anderen Disketten können in die Dateien Startup-Sequence oder StartupII aufgenommen werden. Wenn Sie z. B. den Neustart mit der Extras-Diskette im Laufwerk df1: durchführen, können Sie einen Pfad dorthin zusätzlich festlegen:

```
path  Extras1.3D:  add
```

Wenn Sie den Neustart immer mit einer Diskette (hier als Beispiel "Meine" genannt) durchführen wollen, die C-Programmtexte und die Verzeichnisse "Includes" und "Lib" enthält, kann es komfortabel sein, logische Namen für diese Verzeichnisse festzulegen. Wenn sich Ihre andere Befehlsdateien immer auf diese logischen Namen beziehen, müssen bei Änderungen des Systems oder der Konfiguration nur diese Zuweisungen geändert werden.

```
assign  CPROGS:  Meine:
assign  INCLUDE:  Meine:include
assign  LIB:  Meine:lib
```

5.3 Anmelden von Festplatten-Partitionen

Nach der Ausführung des Befehls BindDrivers können weitere Festplatten-Partitionen angelegt, alle Systemverzeichnisse den Festplattenverzeichnissen zugewiesen und zusätzliche Pfade zu den persönlichen Verzeichnissen in den Festplatten-Partitionen definiert werden. Eine flexible Methode, diese Abläufe ausführen zu lassen ist, alle zusätzlichen Befehle in einer "HD-Startup" genannten Befehlsdatei zusammenzufassen und diese Datei im Verzeichnis S: in der ersten Partition zu plazieren. Fügen Sie dann in der Datei Startup-Sequence nach dem Befehl BindDrivers die Zeile "Execute dh0:s/hd-startup" ein.

Wechseln Sie vor der Ausführung der Befehlsdatei HD-STARTUP durch CD nach RAM:, falls der Befehl EXECUTE eine temporäre Datei anlegen muß. Falls Sie Ihre Diskette für den Neustart mit mehr als einem Amiga-System benutzen, werden diese (nach dem Befehl BindDrivers eingefügten) Zeilen nur dann die Datei HD-STARTUP ausführen, wenn der Neustart mit dem Festplatten-System erfolgt. (Abhängig von Ihrer Konfiguration kann es erforderlich sein, "dh0:" durch einen anderen Einheitennamen zu ersetzen.)

```
cd ram:
failat 30
assign >NIL: dh0: exists
if not warn
  execute dh0:s/hd-startup
  endif
failat 10
```

5.4 RAM: umbenennen

Bearbeiten Sie die Datei StartupII und fügen Sie nach der Zeile "assign CLIPS: ram:clipboards" (Zeile 15) die folgende Zeile ein:

```
relabel drive RAM: name RAMDISK
```

5.5 Weitere Befehle resident laden

Die meisten der Befehle im Verzeichnis C: und das Programm MORE der Version 1.3 können resident geladen werden (erkennbar durch das Bit "p" [Pure], das beim AufLISTen dieser Befehle angezeigt wird). Resident geladene Befehle werden extrem schnell ausgeführt, erfordern beim Verarbeiten mehrerer Tasks gleichzeitig (Multitasking) weniger Speicherplatz und gestalten das Arbeiten mit der SHELL im allgemeinen angenehmer. Letzteres gilt besonders bei auf Disketten basierenden Systemen.

Ein Vorschlag für das residente Laden zusätzlicher Befehle ist die Erstellung einer separaten Befehlsdatei mit dem Namen "rescom", ähnlich dem folgenden Beispiel:

```
resident c:dir pure
resident c:copy pure
resident c:delete pure
resident c:ed pure
resident sys:Utilities/More pure
```

Plazieren Sie die Datei im Verzeichnis S:. Bearbeiten Sie dann die Datei s:StartupII und fügen Sie nahe dem Ende folgende Zeile ein:

```
run >NIL: execute s:RESCOM
```

Da es nur eine Liste der residenten Befehle gibt, die für das gesamte System gilt, kann die durch RUN und EXECUTE gestartete Verarbeitung der Datei RESCOM ablaufen und können die Befehle resident geladen werden, während die Verarbeitung der Datei StartupII beendet wird.

5.6 Umgebungsvariablen festlegen

Das Programm MORE der Version 1.3 erlaubt Ihnen nun, wenn es über den CLI oder die SHELL gestartet wird, die angezeigte Datei zu bearbeiten. Es kann dazu jeder Editor verwendet werden, an den ein Dateiname als einzelnes Argument einer Befehlszeile übergeben werden kann. MORE lädt den Namen des von Ihnen bevorzugten Editors aus der Umgebungsvariablen EDITOR. Soll diese Funktion von MORE verwendet werden, sollte eine Zeile in die Datei StartupII eingefügt werden. In dieser wird durch den Befehl SETENV der Umgebungsvariablen EDITOR der Pfad und der Name des zu verwendenden Editors zugewiesen. Fügen Sie die Zeile mit SETENV nach der Zeile "assign ENV: RAM:ENV" (Zeile 13) ein.

```
SETENV EDITOR C:ed ; (oder ein anderer von Ihnen gewünschter
Editor)
```

5.7 Eigene Befehlsmakros in die Datei S:SHELL-STARTUP aufnehmen

Unter Verwendung eines Texteditors fügen Sie folgende Makro-Definitionen in die Datei Shell-Startup ein.

Beispiele:

```
alias ns newshell
alias es endshell
alias ex execute
alias ls list
alias prefs sys:prefs/Preferences
```

5.8 Anlegen einer resetfesten RAM-Disk (RAD:)

Die Datei MountList im Verzeichnis DEVS: der Workbench-Diskette, Version 1.3, enthält einen Eintrag für RAD:, eine resetfeste RAM-Disk, von der aus ein Neustart durchgeführt werden kann, wenn die Kickstart-Version 1.3 verwendet wird. Die Kapazität von RAD: kann durch Änderung des Wertes für "HighCyl" im RAD:-Eintrag in der Datei MountList festgelegt werden. Vor dem Versuch, die resetfeste RAM-Disk mit der neuen Kapazität anzumelden, muß ein Kaltstart nach Ausschalten des Rechners durchgeführt werden.

Bei Amiga-Systemen mit 1 MByte Kapazität und ohne Festplattenlaufwerk kann eine kleine RAD: verwendet werden, um das Verzeichnis S: und einige häufig benutzte Befehle aus dem Verzeichnis C: zu speichern. Ist Ihr System mit mehr als 2 MByte RAM ausgestattet, kann eine resetfeste RAM-Disk mit der Kapazität einer Diskette (880 KByte, HighCyl=79) angelegt werden. Für die Erstellung einer superschnellen, resetfesten "Workbench-Diskette" im RAM kann die Workbench-Diskette zu einer resetfesten RAM-Disk mit 880 KByte kopiert werden. Sie ist auch bei Verwendung von Kickstart 1.3 für den Neustart einsetzbar. Besitzer von Festplattenlaufwerken können, falls gewünscht, eine resetfeste RAM-Disk anlegen, welche die Mindestanzahl von Dateien und Verzeichnissen aufnehmen kann, die für die Übergabe der Steuerung an die Festplatte erforderlich sind.

5.8.1 Inhalt der Workbench-Diskette in die resetfeste RAM-Disk - Für Amiga-Systeme mit mehr als 2 MByte RAM

Die Übernahme des Inhalts der Workbench-Diskette in die resetfeste RAM-Disk ist extrem nützlich, wenn Sie 1 MByte RAM aufwenden können und nicht über ein Festplattenlaufwerk verfügen. Beim Kaltstart werden die nachstehend beschriebenen Änderungen eine resetfeste RAM-Disk mit 880 KByte Kapazität anmelden und den Inhalt der Startdiskette durch DISKCOPY dorthin kopieren. Dies erfordert etwa ein Viertel der Zeit, die für das Kopieren von Diskette zu Diskette benötigt wird und muß nicht bei jedem Neustart wiederholt werden. Alle Systemverzeichnisse werden der resetfesten RAM-Disk zugewiesen, so daß der Rest der Dateien Startup-Sequence und StartupII sehr schnell ausgeführt wird und das Laufwerk df0: für die Arbeit frei zur Verfügung steht.

In Verbindung mit der Kickstart-Version 1.3 können Sie den Neustart unter Verwendung der entstandenen resetfesten RAM-Disk durchführen. Schalten Sie den Amiga nach Durchführung der folgenden Änderungen mindestens 30 Sekunden aus. Schalten Sie ihn dann wieder ein, und starten Sie ihn unter Verwendung der veränderten Workbench-Diskette. Beachten Sie, daß nach dem Kopieren von df0: zu RAD: (durch Befehl DISKCOPY) zweimal ein Kommunikationsfenster aufblinkt. Das ist normal.

1. Bearbeiten Sie die Datei DEVS/MountList und ändern Sie den Wert von HighCyl im Eintrag RAD: zu "HighCyl=79" (derselbe Wert, wie der für eine Diskette mit 880 KByte Kapazität).
2. Fügen Sie die folgenden Zeilen in der Datei Startup-Sequence ein und zwar zwischen den Zeilen mit den Befehlen "FastMemFirst" und "BindDrivers":
   failat 30

```
assign >NIL: RAD: exists
if warn
 echo "Anmelden von RAD:"
 mount RAD:
 if not exists RAD:c
  echo "Kopieren von df0: zu RAD:"
  SYS:System/diskcopy <NIL: df0: to RAD: name "RAMWB"
  endif
 endif
RAD:c/failat 10

RAD:c/cd RAD:c

RAD:c/echo "Übergabe der Steuerung an RAD: ..." NOLINE
assign c: RAD:c
assign s: RAD:s
assign l: RAD:l
assign libs: RAD:libs
assign devs: RAD:devs
assign fonts: RAD:fonts
assign sys: RAD:
echo "beendet"
```

5.8.2 Kleinste resetfeste RAM-Disk für Übergabe der Steuerung an eine Festplattensteuerung A2090 und das Festplattenlaufwerk

Eine kleine, resetfeste RAM-Disk kann verwendet werden, um die Steuerung an ein Festplattenlaufwerk zu übergeben. Dies erlaubt Benutzern mit Festplattensteuerungen, die nicht für den Neustart verwendet werden können (A2090), einen Neustart durchzuführen, ohne die Workbench-Diskette in Laufwerk df0: einzulegen. Die resetfeste RAM-Disk muß Ihre Dateien DEVS/SYSTEM-CONFIGURATION, S:STARTUP-SEQUENCE und alle erforderlichen Dateien und Verzeichnisse enthalten, die für folgende Abläufe erforderlich sind: Ausführung des Befehls BindDrivers und Zuweisung der Steuerung an die Partition der Festplatte, die die Workbench-Dateien und -Verzeichnisse enthält.

Für diesen Start ist eine Anfangspartition erforderlich, die mindestens die Dateien und Verzeichnisse enthält, die für die Ausführung der Datei HD-STARTUP benötigt werden. Die Datei HD-STARTUP meldet alle weiteren Partitionen an und weist die logischen System-Verzeichnisse der/den Partition(en) zu, die die Workbench-Verzeichnisse enthalten. Es sind auch zusätzliche Dateien aufgelistet, die in dieser Anfangspartition erforderlich sind, wenn die Festplattensteuerung A2090A mit der Fähigkeit des automatischen Neustarts eingesetzt wird. (Sie können diese Dateien sofort dort unterbringen oder einen Bereich freihalten, um sie später anzulegen.) Beachten Sie, daß diese Dateien nur dazu vorgesehen sind, die Steuerung an eine andere Partition (wahrscheinlich mit dem Dateisystem FFS) zu übergeben, die alle Workbench-Verzeichnisse und -Befehle enthält. Wenn der Start ausgehend von der Anfangspartition durchgeführt werden soll, können Sie ebensogut den Inhalt der gesamten Workbench-Diskette dorthin kopieren.

1. Stellen Sie sicher, daß die Anfangspartition der Festplatte die folgenden Verzeichnisse, Dateien und Befehle enthält:

s (Verzeichnis)	
startup-sequence	(autom. Ausführung beim Neustart)
hd-startup	
c (Verzeichnis)	
cd	
assign	
echo	
mount	
BindDrivers	(autom. Ausführung beim Neustart)
AddBuffers	(autom. Ausführung beim Neustart)
SetPatch	(autom. Ausführung beim Neustart)
devs (Verzeichnis)	
MountList	
system-configuration	(autom. Ausführung beim Neustart)
libs (Verzeichnis)	
icon.library	(autom. Ausführung beim Neustart)
l (Verzeichnis)	
disk-validator	
FastFileSystem	(beim Anmelden von FFS-Partitionen)
System (Verzeichnis)	
FastMemFirst	(autom. Ausführung beim Neustart)
Expansion (Verzeichnis)	
andere Treiber ohne die Fähigkeit für den automatischen Neustart	(autom. Ausführung beim Neustart)

2. Kopieren Sie den Festplattentreiber A2090 und das Piktogramm (HdDisk und HdDisk.info) zu zwei Zielen: zur Schublade Expansion auf Ihrer Workbench-Diskette und zur Schublade Expansion in jener Festplattenpartition, die Verzeichnis SYS: (das alle Workbench-Verzeichnisse enthält) sein wird. Im folgenden Beispiel ist das die Partition FF0:, welcher der Name FASTWB: zugewiesen wird.

3. Verändern Sie den Eintrag für RAD: in der Datei devs/MountList, indem Sie den Wert für HighCyl von 21 zu 7 ändern. Fügen Sie die Einträge für alle weiteren vorhandenen Partitionen der Datei MountList hinzu.

4. Fügen Sie die folgenden Zeilen direkt nach der Zeile mit dem Befehl BindDrivers (Zeile 6) in die Datei Startup-Sequence ein. Abhängig von der Konfiguration Ihres Systems kann es erforderlich sein, als Verweis zur Anfangspartition eine andere Partition als dh0: einzugeben.

```
failat 30
assign >NIL: INIT: dh0:
if not warn
 cd INIT:
 execute INIT:s/hd-startup
 endif
failat 10

; Nun müssen die Workbench-Funktionen von der Festplatte ausgehend
ablaufen.
; Eine resetfeste RAM-Disk (RAD:) für das wiederholte Neustarten erstellen.
assign >NIL: RAD: exists
if warn
 echo "Anmelden von RAD: ..." NOLINE
 mount RAD:
 if not exists RAD:c
   echo "Vorbereitung ..." NOLINE
   relabel drive RAD: name RamDrive
   makedir RAD:c
   copy c:AddBuffers|assign|BindDrivers|cd|copy|echo RAD:c quiet
   rad:c/copy c:endif|if|execute|failat|makedir|SetPatch RAD:c quiet
   rad:c/makedir RAD:devs
   rad:c/copy devs:system-configuration RAD:devs quiet
   rad:c/makedir RAD:L
   rad:c/copy L:disk-validator RAD:L quiet
```

```
    rad:c/makedir RAD:Libs
    rad:c/copy Libs:icon.library RAD:Libs quiet
    rad:c/makedir RAD:System
    rad:c/copy SYS:System/FastMemFirst RAD:System quiet
    rad:c/makedir RAD:s
    rad:c/copy s:startup-sequence RAD:s quiet
    rad:c/makedir RAD:Expansion
    rad:c/copy sys:Expansion rad:Expansion quiet
    endif
  echo "beendet"
  endif
```

5. Erstellen Sie die Datei HD-STARTUP und speichern Sie sie im Verzeichnis S in der Anfangspartition der Festplatte (hier durch den Namen INIT: bezeichnet, der in der oben aufgeführten Erweiterung der Startsequenz zugewiesen wurde). Die Datei HD-STARTUP wird Ihre weiteren Partitionen anmelden, die Steuerung an sie übergeben und Ihre Festplattenpfade und logischen Zuweisungen definieren. Nachstehend ist ein Beispiel für eine Datei HD-STARTUP aufgeführt. Sie kann für die Anpassung an Ihr Festplattensystem verändert werden.

```
; Zuweisung der Systemverzeichnisse für die Anmeldung der restlichen
; Partitionen.
INIT:c/cd INIT:c
assign DEVS: INIT:Devs
assign L: INIT:L
; In diesem Beispiel verfügt der Benutzer über 2 FFS-Paritionen, FF0: und FF1:
; FF0: enthält alle Workbench-Verzeichnisse. Wir weisen den logischen
; Namen FASTWB: der Partition zu, die die Workbench-Verzeichnisse
;enthält und den Namen H: der Arbeitspartition des Benutzers,
;um das Ändern von ;Befehlsdateien zu erleichtern.
mount FF0:
mount FF1:
assign FASTWB: FF0:
assign H: FF1:
; Zuweisung aller Systemverzeichnisse an FASTWB:
assign C:     FASTWB:C
assign SYS:   FASTWB:
assign DEVS:          FASTWB:Devs
assign LIBS:  FASTWB:Libs
assign L:     FASTWB:L
assign S:     FASTWB:S
assign FONTS:         FASTWB:Fonts
```

6. Kopieren Sie nun die geänderte Datei MountList in das Devs-Verzeichnis der Anfangspartition und zur Workbench-Partition. Kopieren Sie die geänderte Datei Startup-Sequence in das Verzeichnis S der Anfangspartition und der Workbench-Partition. (Sie wird von der Workbench-Partition zu RAD: kopiert werden.) Schalten Sie das System für mindestens 30 Sekunden aus und führen Sie den Neustart unter Verwendung der geänderten Workbench-Diskette durch.

5.8.3 Kleine resetfeste RAM-Disk für Amiga-Systeme mit 1 MByte RAM

Eine kleine resetfeste RAM-Disk ist nicht für die Durchführung des Neustarts geeignet, da sie die beim Neustart erforderlichen Befehle und Handler nicht aufnehmen kann. Sie ist jedoch teilweise nützlich, um das Laufwerk df0: für die Arbeit freizuhalten, da einige häufig verwendete Befehle, das Verzeichnis S: und das Verzeichnis Expansion dort untergebracht werden können. Mit dieser resetfesten RAM-Disk sind Sie auch in der Lage, durch den Befehl NEWSHELL in der Befehlszeile oder das Piktogramm in der resetfesten RAM-Disk eine neue SHELL zu öffnen, ohne die Workbench-Diskette einzulegen.

Nach der Durchführung dieser Veränderung dauert der Start einige Minuten, wenn Sie das System einschalten. Die nächsten Neustarts laufen jedoch bedeutend schneller ab.

1. Öffnen Sie das Workbench-Diskettenpiktogramm und öffnen Sie dann die Schublade Prefs. Klicken Sie einmal auf das Piktogramm Pointer und wählen Sie INFO aus dem Workbench-Menü. Ändern Sie den Wert von Default Tool von "SYS:Prefs/Preferences" zu "Preferences". Drücken Sie die Taste RETURN und klicken Sie auf SAVE, um die Änderung zu speichern. Wiederholen Sie diesen Ablauf für die Piktogramme "Printer" und "Serial".

2. Führen Sie nun die folgenden Änderungen in der Datei devs/MountList durch:
   - Fügen Sie im Eintrag NEWCON: diese Zeile hinzu: Mount = 1
   - Ändern Sie im Eintrag RAD: den Wert von HighCyl von 21 zu 14, und fügen Sie diese Zeile hinzu: BootPri = -129

3. Entfernen Sie aus der Datei Startup-Sequence die folgenden Zeilen (Zeile 18 und 19), die sich vor der Zeile mit dem Befehl LoadWB befinden:

```
18 SYS:System/SetMap d ; Deutsche Tastaturbelegung
aktivieren
19 path ram: c: sys:utilities sys:system s: sys:prefs
add
```

4. Fügen Sie die folgenden Zeilen in die Datei Startup-Sequence ein und zwar zwischen den Zeilen mit den Befehlen FastMemFirst und BindDrivers (Zeile 5 und 6). Sie können die Zeilen mit dem Befehl COPY im folgenden Beispiel ändern, um andere Befehle in die resetfeste RAM-Disk zu kopieren. Die Verzeichnisse und einige der Befehle (wie NEWSHELL, ENDCLI, RUN, FORMAT und DISKCOPY) sind jedoch erforderlich.

```
failat 30
assign >NIL: RAD: exists
if warn
  echo "Anmelden von RAD: . . ." NOLINE
  mount RAD:
  if not exists RAD:c
   echo "Vorbereitung von RamDrive ..." NOLINE
   relabel drive RAD: name RamDrive
   makedir RAD:c
   copy c:assign|copy|delete|dir|echo|ed|else RAD:c quiet
   rad:c/copy c:endcli|endif|if|info|makedir RAD:c quiet
   rad:c/copy c:newshell|path|prompt|rename|run RAD:c quiet
   rad:c/copy sys:Utilities/More RAD:c quiet
   rad:c/makedir RAD:s
   rad:c/copy s: RAD:s all quiet
   rad:c/makedir RAD:System
   rad:c/copy sys:System/CLI|format|diskcopy RAD:System quiet
   rad:c/copy sys:Shell|Shell.info RAD: quiet
   rad:c/makedir RAD:Expansion
   rad:c/copy sys:expansion rad:Expansion quiet
   endif
  endif
failat 10
assign s: RAD:s
path rad:c rad:system sys:utilities sys:system sys:prefs sys: s: add
Sys:System/SetMap d
assign sys: RAD:
echo "beendet"
```

5. Fügen Sie vor der Zeile mit dem Befehl LoadWB diese beiden Zeilen ein:

```
path c: ram: add
assign c: rad:c
```

6. Hilfsprogramme auf der Extras-Diskette

6.1 Programme

Die Schublade Tools enthält einige Hilfsprogramme, die Sie mit Ihrem Amiga "arbeiten" lassen. Das bedeutet, daß sie die grundlegenden Fähigkeiten des Arbeitstisches erweitert. Mit diesen Programmen können Sie die Piktogramme und Zeichensätze modifizieren, die Speicherbelegung des Systems prüfen, Textdateien bearbeiten und die Farben des Bildschirms ändern. Die Schublade Tools enthält:

FED — Änderung von vorhandenen Zeichensätzen oder Neuerstellung
FreeMap — Anzeige der Belegung des CHIP-Speichers
IconEd — Änderung von Piktogramm-Bildern
IconMerge — Mischen und Trennen von Doppel-Piktogrammen
KeyToy2000 — Anzeige der kompletten Tastaturbelegung des Amiga 2000/500
MEMACS — Edierung von Textdateien
Palette — Änderung der Farben eines Bildschirms
PerfMon — Anzeige der Leistung des Systems

Jedes dieser Hilfsprogramme wird nachstehend beschrieben, mit Ausnahme von MEMACS. MEMACS ist ein so leistungsstarker und komplexer Texteditor, daß er ein eigenes Kapitel erfordert. Er wird in Kapitel 7 umfassend behandelt.

6.1.1 FED

FED ist ein Zeichensatz-Editor für die Änderung der Workbench-Zeichensätze oder die Erstellung eigener, neuer Zeichensätze. Wenn Sie Änderungen eines Workbench-Zeichensatzes vornehmen wollen, sollten Sie dazu unbedingt eine Kopie der Workbench-Diskette verwenden. Nach durchgeführten Änderungen kann der Originalzustand nicht wieder zurückgerufen werden. Bitte beachten Sie, daß mit FED keine Farb-Zeichensätze oder Zeichensätze, bei denen eine Dimension mehr als 32 Bildpunkte umfaßt, bearbeitet werden können.

Wenn Sie auf das FED-Piktogramm doppelklicken, wird ein Fenster geöffnet, das Symbole, Felder, Schalter, Zahlen und Zeichensymbole enthält. Am oberen Rand ist auch eine Menüleiste vorhanden, deren Menüs die Bezeichnungen Project, Edit und Attributes tragen. Alle Abläufe der Erstellung/Bearbeitung werden in diesem Fenster durchgeführt. Nachstehend werden alle vorhandenen Menübefehle und Symbole beschrieben.

Wenn Sie einen Workbench-Zeichensatz bearbeiten wollen, muß dieser zuerst in das Programm FED geladen werden. Wählen Sie dazu Open aus dem Menü Project. Es erscheint ein Kommunikationsfenster, in dem alle verfügbaren Zeichensätze aufgelistet sind. Wählen Sie den gewünschten aus und wählen Sie Load It. Falls Sie Ihre Meinung geändert haben, wählen Sie Whoops! (Soll ein völlig neuer Zeichensatz erstellt werden, muß im Menü Project der Befehl New anstelle von Open gewählt werden.)

Jedes Zeichen des Zeichensatzes erscheint in einem der Zeichenfelder an der rechten Seite des Bildschirms. Die Anzahl der gleichzeitig angezeigten Felder ist begrenzt, da ein normaler Zeichensatz mehr als 200 Felder mit Zeichen enthält. Das Zeichenfeld zeigt die aktuelle Darstellung eines Zeichens. Sobald Änderungen des Zeichens vorgenommen werden, ändert sich auch die Darstellung im Zeichenfeld.

Jedes Zeichen besteht aus einer Anzahl von Punkten. Jeder Punkt stellt einen Bildpunkt des Zeichens dar. Für die Bearbeitung (oder Erstellung) eines Zeichens im großen Zeichenfeld, das die linke Seite des Bildschirms ausfüllt, wird die Maus verwendet.

- Für die Änderung eines Punktes (gedruckt/nicht gedruckt) wird er mit der Maus angeklickt.
- Für das Zeichnen von Punktlinien klicken Sie auf einen nicht vorhandenen Punkt und ziehen die Maus.
- Für das Löschen von Punktlinien klicken Sie auf einen vorhandenen Punkt und ziehen die Maus.

Jedes der Symbole wird nachstehend kurz erklärt, so daß Sie deren Funktionen verstehen. Dann sollten Sie sich von Ihrer Kreativität und Vorstellung leiten lassen, um außergewöhnliche Zeichensätze zu erstellen.

Die Symbole

Schalter LoRes — Wählt die Auflösung 320 x 200 oder 640 x 200 Punkte.
Schalter Grid — Schalter das Raster im Edierfeld ein und aus.

Roll-Raute — Ist ein Zeichen größer als der Anzeigebereich des Edierfensters, können die außerhalb liegenden Teile in diesen Bereich gerollt werden, indem auf den jeweiligen Richtungspfeil in der Roll-Raute geklickt wird. Das Zeichen bewegt sich nur dann, wenn seine äußerste Kante noch nicht im Anzeigebereich liegt. Ein Zeichen kann zum Beispiel nicht nach links rechts bewegt werden, wenn seine linke Kante bereits zu sehen ist. Dies gilt auch dann, wenn die Zeichenkante nicht an der Kante des Edierfelds liegt.

Zoom — Erlaubt die Anzeigegröße des Edierfelds zu wählen.

Dreieck nach links/rechts — Verschiebt das aktuelle Zeichen um einen Bildpunkt nach links oder rechts.

Dreieck nach oben/unten — Verschiebt das aktuelle Zeichen um einen Bildpunkt nach oben oder unten.

R — Für inverse Darstellung des Zeichens

Pfeil nach links/rechts — Spiegelverkehrte Darstellung des Zeichens nach links/rechts.

Pfeil nach oben/unten — Spiegelverkehrte Darstellung des Zeichens nach oben/unten.

Die Felder

Durch die Felder werden spezielle Parameter für den Zeichensatz festgelegt. Weitere Erklärungen zu diesen Parametern sind im Kapitel "Text" des Handbuchs "ROM Kernel Manual: Libraries and Devices" von Addison-Wesley zu finden.

Die Felder Lo und Hi bestimmen die Anzahl der Zeichen in einem Zeichensatz oder den Zeichenbereich (d. h. welche Zeichen des ASCII-Codes gespeichert werden sollen). Sie können z. B. einen Bereich erstellen, der nur Großbuchstaben (065 bis 090) oder einen, der nur numerische Symbole (048 bis 057) enthält. Der Bereich für die Zeichensätze auf der Workbench-Diskette reicht von 032 bis 255.

Kern — Bestimmt den Grad des Kerning (Überlappung von Zeichen, z. B. "e" wird teilweise unter dem Querstrich des "T" geschrieben). Dieser Wert ist normalerweise mit 0 angegeben.

Space — Bestimmt die Informationen für proportionalen Zeichenabstand oder die Breite jedes Zeichenrechtecks - wie viele Bildpunkte das Zeichen auf dem Bildschirm einnimmt.

Width — Arbeitet zusammen mit der Variablen "Space", um die Informationen bezüglich des proportionalen Zeichenabstands festzulegen - wie viele Bildpunkte das eigentliche Zeichen umfaßt.

Die Felder Kern, Space und Width können nur bei Zeichensätzen mit Proportionalschrift verändert werden.

BLine — Legt die Position der Grundlinie des Zeichens fest. Die Grundlinie wird durch eine grüne, waagerechte Linie im Edierfeld dargestellt. Es wird die Anzahl der Rasterblöcke zwischen dem oberen Rand des Zeichens und der Grundlinie angegeben. Die Grundlinie ist mit einem Strich auf einem linierten Blatt Papier vergleichbar.

XSize — Stellt die Nennbreite des Zeichensatzes dar.

YSize — Stellt die Nennhöhe des Zeichensatzes dar.

Das Menü Project

New — Löscht die Zeichenfelder und erlaubt die Neuerstellung eines Zeichensatzes.

Open —, Zeigt eine Liste der verfügbaren Zeichensätze und ermöglicht, daß einer davon geladen wird.

Save — Speichert den Zeichensatz, der momentan bearbeitet wird, unter seinem ursprünglichen Namen.

Save-as — Erlaubt das Speichern des Zeichensatzes unter einem neuen Namen.

About — Zeigt einen Autorenvermerk des FED-Programms.

Quit — Beendet FED.

Das Menü Edit

HINWEIS: Die Befehle in diesem Menü beeinflußen den GESAMTEN Zeichensatz, und einige davon sind sehr schwer rückgängig zu machen.

Make italic — Algorithmische Umstellung des gesamten Zeichensatzes zu Kursivschrift.

Make bold — Algorithmische Umstellung des gesamten Zeichensatzes zu Fettschrift.

Make underlined — Algorithmische Umstellung des gesamten Zeichensatzes zu unterstrichener Schrift.

Copy to — Kopiert ein Zeichen von einem Zeichenfeld in ein anderes. Soll dieser Befehl verwendet werden, ist ein Zeichenfeld auszuwählen, zum Beispiel K, und dann der Befehl Copy to zu wählen. Wenn Sie nun ein anderes Zeichenfeld auswählen, wird "K" in dieses Feld kopiert. Solange Copy to gewählt ist, wird jedes weitere ausgewählte Feld mit einer Kopie des anfänglich gewählten versehen. Der Modus Kopieren bleibt aktiv, bis Copy to erneut gewählt wird.

Erase — Löscht das Zeichen aus dem ausgewählten Zeichenfeld.

All right — Verschiebt alle Zeichen um einen Bildpunkt nach rechts.

All left — Verschiebt alle Zeichen um einen Bildpunkt nach links.

All up — Verschiebt alle Zeichen um einen Bildpunkt nach oben.

All down — Verschiebt alle Zeichen um einen Bildpunkt nach unten.

Das Menü Attributes

Durch dieses Menü können Sie die Kennzeichnungsbits für Typ und Schriftart des Zeichensatzes festlegen.

Font Type - Wählen Sie entweder proportionale oder für alle Zeichen gleichbleibende Zeichenbreite.

Font Style - Wählen Sie Normal, Italic (kursiv), Bold (fett), Underlined (unterstrichen) oder Extended (erweitert). Setzen Sie für die meisten Zeichensätze "Normal". Wenn Sie z. B. einen speziellen Zeichensatz in Kursivschrift erstellen wollen, wählen Sie

"Italic". Findet das System einen speziell in Kursivschrift vorhandenen Zeichensatz, benutzt es diesen, statt zu versuchen, einen eigenen zu erstellen.

6.1.2 FreeMap

Das Hilfsprogramm FreeMap bewirkt die graphische Darstellung der CHIP-Speicherbelegung, indem die verwendeten Blöcke und die freien Bereiche gezeigt werden. Jeder Punkt im Darstellungsbereich entspricht einem Block von 64 Bytes. Sind alle Bytes in einem Block frei, wird der Punkt dunkel angezeigt. Ist ein Byte belegt, erscheint der Punkt hell. Am unteren Rand des Fensters wird der Umfang des freien CHIP RAM und FAST RAM angegeben.

Für das Verlassen von FreeMap wählen Sie Please aus dem Menü Quit. Wird Please aus dem Menü Info gewählt, erscheint eine kurze Erklärung zu FreeMap.

6.1.3 IconEd

Das Hilfsprogramm IconEd erlaubt, die Bilddarstellung der Piktogramme auf dem Arbeitstisch zu verändern. Es wurde aus der Systemschublade des Arbeitstischs entfernt und ist nun auf der Extras-Diskette untergebracht. Das Programm IconEd ist im Benutzerhandbuch des Amiga ausführlich beschrieben.

6.1.4 IconMerge

Durch das Programm können aus zwei separaten Abbildungen zweiteilige Piktogramme für den Arbeitstisch erstellt werden. Es erlaubt auch die Trennung der zweiteiligen Abbildung eines Piktogramms in zwei separate Abbildungen. Bei einem Piktogramm mit zweiteiliger Abbildung wird ein ganz anderes Bild angezeigt, wenn es gewählt ist.

Durch Doppelklicken auf das Piktogramm öffnet sich das IconMerge-Fenster, das Sie dazu auffordert, entweder ein "m" einzugeben, um zwei Piktogramme mit je einer Abbildung zu mischen, oder ein "s", um ein Piktogramm mit doppelter Abbildung in zwei separate zu trennen. Sollen z. B. die Piktogramme von ICONED und FED gemischt werden, ist als Antwort auf die Eingabeaufforderung ein "m" anzugeben, worauf folgende Meldung erscheint:

```
MERGE:  Icon1,  Icon2  (merge)->  Icon3
Enter all icon filenames without ".info".
Icon1:
```

Geben Sie nach der Eingabeaufforderung Icon1: den Namen "IconEd" ein und drücken Sie RETURN. Es erscheint eine zweite Eingabeaufforderung, die nach "Icon2" fragt; geben Sie "FED" ein, und drücken Sie RETURN. Geben Sie für "Icon3" den Namen Test ein. Nach dem Drücken der Taste RETURN mischt das Programm IconMerge die beiden Piktogramme, und das Fenster verschwindet.

Für die Anzeige des neuen Piktogramms muß zuerst das Fenster Tools geschlossen und erneut geöffnet werden. Das Piktogramm Test liegt über dem Piktogramm IconEd. Ziehen Sie einfach das Piktogramm Test in einen leeren Bereich des Fensters. Beachten Sie, daß das Test-Piktogramm eine exakte Kopie des IconEd-Piktogramms ist (Icon1). Wenn Sie auf das Test-Piktogramm klicken, erscheint das FED-Piktogramm (Icon2), um zu zeigen, daß das Piktogramm gewählt ist.

Das Piktogramm des Programms IconMerge besteht aus zwei Abbildungen. Versuchen Sie, diese beiden in separate Abbildungen zu trennen. Doppelklicken Sie auf das Piktogramm "IconMerge", um das IconMerge-Fenster zu öffnen. Antworten Sie auf die Frage nach Mischen oder Separieren mit "s" und die folgende Anzeige erscheint:

```
SPLIT:  Icon1  (split)->  Icon2,  Icon3
Enter all icon filenames without ".info":
Icon1:
```

Geben Sie nach der Eingabeaufforderung Icon1: den Namen "IconMerge", für Icon2: den Namen "Icon" und für Icon3: den Namen "Merge" ein. Nach dem Verschwinden des Fensters schließen Sie das Fenster Tools und öffnen es wieder. Die beiden neuen Piktogramme, Icon und Merge, liegen hinter dem Original-Piktogramm IconMerge.

Ziehen Sie sie in einen leeren Bereich des Fensters (oder verwenden Sie den Befehl Clean-up aus dem Menü Special des Arbeitstischs). Icon zeigt das Aussehen des Piktogramms IconMerge, solange dieses nicht gewählt ist, Merge die Darstellung, wenn es gewählt ist.

Wenn Sie eines der beiden neuen Piktogramme wählen, werden sie nur hervorgehoben dargestellt. Die Form wird nicht geändert, wie es bei einem Piktogramm mit Doppelabbildung erfolgt. Das Aufteilen von Doppelabbildungen in zwei separate Abbildungen ermöglicht, daß jedes von beiden unter Verwendung von IconEd bearbeitet wird. Die veränderten Abbildungen können dann durch die Option "m" von IconMerge wieder zusammengefügt werden.

Soll das Programm IconMerge über den Befehlsprozessor gestartet werden, lautet das Befehlsformat wie folgt:

```
ICONMERGE  [<icon1> <icon2> <icon3> [OPT S]]
```

Die Standardfunktion von IconMerge ist das Mischen von zwei Piktogrammen. Werden keine Piktogramm-Namen angegeben, erscheint das Fenster IconMerge, und Sie können wie zuvor beschrieben fortfahren. Wird OPT S angegeben, werden zwei Piktogramme mit den Namen <Icon2> und <Icon3> erstellt.

Einige Dinge, die zu beachten sind:

1) Piktogramm-Namen müssen ohne die Erweiterung ".info" eingegeben werden, aber ggf. mit Pfad vorweg.
2) Stellen Sie beim Mischen von Piktogrammen sicher, daß beide dieselbe Größe aufweisen.

6.1.5 KeyToy2000

Doppelklicken auf das Piktogramm KeyToy2000 bewirkt, daß die momentan gültige, vollständige Tastenbelegung der Tastatur angezeigt wird, die mit dem Amiga 2000 bzw. Amiga 500 geliefert wird.

In der ersten Anzeige werden die Zeichen gezeigt, die ausgegeben werden, wenn eine der Tasten einzeln gedrückt wird. Für die Anzeige der Zeichen, die ausgegeben werden, wenn eine der Shift- oder Alt-Tasten oder die Taste Control gedrückt wird, klicken Sie entweder auf das entsprechende Symbol in der KeyToy-Anzeige, oder betätigen Sie die Taste auf der Tastastur.

Die folgende Liste soll als Leitfaden für die KeyToy-Anzeige dienen:

- Alle KeyToy-Tasten, die rot gekennzeichnet sind, werden nicht durch KeyToy verwendet. (Deshalb erscheinen in der Anfangsanzeige die Tasten CTRL, SHIFT und ALT blau. Wird eine oder werden mehrere dieser Tasten gewählt, ändern die Kennzeichnungen ihre Farbe.)
- Alle rot gekennzeichneten Tasten sind Akzenttasten (ohne Cursorschritt), mit Ausnahme der AMIGA-Tasten und der Taste CAPS LOCK.
- Alle fett und kursiv gekennzeichneten Tasten können mit Akzenten versehen werden.
- Alle mit $$ gekennzeichneten Tasten bewirken eine Zeichenfolge, die länger ist als ein Zeichen.

- Alle Zeichen, denen ein ^ oder ~ vorangestellt ist, sind Steuerzeichen.
- Wird eine Taste leer dargestellt, bewirkt sie in Verbindung mit der/den momentan aktiven Steuertaste(n) (CTRL, SHIFT, ALT) keine Zeichenausgabe.

6.1.6 Palette

Durch das Programm Palette können die Darstellungsfarben eines Bildschirms (Screens) verändert werden. Es unterscheidet sich von der Möglichkeit der Farbwahl über das Voreinstellerprogramm Preferences dahingehend, daß der Voreinsteller auf die Farben des Arbeitstisches beschränkt ist. Die mit PALETTE durchgeführten Farbänderungen sind jedoch nur vorübergehend. Sie können nicht auf Diskette gespeichert werden.

Für das Öffnen von PALETTE über den Arbeitstisch ist auf sein Piktogramm doppelzuklicken. Das Fenster des Programms PALETTE erscheint im vordersten Bildschirm, normalerweise dem Arbeitstisch. Sollen die Farben eines anderen Bildschirms verändert werden, muß dieser Bildschirm vor dem Öffnen von PALETTE in den Vordergrund gebracht werden. Öffnen Sie den gewünschten Bildschirm und schieben Sie ihn dann nach unten, damit der Arbeitstisch ebenfalls sichtbar ist. Öffnen Sie das Fenster Tools auf dem Arbeitstisch und doppelklicken Sie auf das Piktogramm PALETTE. Wenn sich das Fenster des Programms PALETTE öffnet, erscheint es als vorderster Bildschirm.

Soll PALETTE durch den Befehlsprozessor gestartet werden, lautet das Befehlsformat wie folgt:

PALETTE [<Bitebenen>] [<Bildschirmart>]

Durch Angabe von Werten für die Optionen <Bitebenen> und <Bildschirmart> kann ein persönlicher Testbildschirm geöffnet werden. Für <Bitebenen> und <Bildschirmart> können die folgenden Werte angegeben werden:

<Bitebenen> Legt die Tiefe des Testbildschirms fest:
1 = 2 Farben, 2 = 4 Farben, 3 = 8 Farben,
4 = 16 Farben und 5 = 32 Farben.

<Bildschirmart> Legt die Auflösung des Testbildschirms fest:
0 = 320 x 200 Bildpunkte, 1 = 320 x 400 Bildpunkte (Zwischenzeilenabtastung), 2 = 640 x 200 Bildpunkte und 3 = 640 x 400 Bildpunkte (Zwischenzeilenabtastung)

HINWEIS: Ist als Wert für <Bildschirmart> entweder 2 oder 3 angegeben, ist der Höchstwert für <Bitebenen> die Ziffer 4.

Das Fenster des Programms PALETTE enthält mehrere Symbole und Regler. Entlang des oberen Fensterrands verlaufen Farbrechtecke, welche die Farben zeigen, die verändert werden können. Unterhalb dieser Farbrechtecke befinden sich drei Farbregler: Rot, Blau und Grün. Wählen Sie die zu verändernde Farbe und verschieben Sie die Regler, bis die gewünschte Farbe erreicht ist. Die Farbänderungen werden in dem senkrechten liegenden Feld, das entlang der Farbregler verläuft, dargestellt. Wiederholen Sie diesen Ablauf für jedes Farbrechteck, bis der Bildschirm wunschgemäß gestaltet ist.

Wählen Sie das Feld OK, um die Farbänderung anzuwenden. Wählen Sie RESET, wenn der Originalzustand der Farben wiederhergestellt werden soll. Oder wählen Sie CANCEL, um das Programm PALETTE zu verlassen, ohne daß Änderungen wirksam werden. ACHTUNG: Wird PALETTE in einem anderen Bildschirm als dem Arbeitstisch geöffnet, muß zuerst PALETTE geschlossen werden, bevor die Anwendung, von der der neue Bildschirm geöffnet wurde, verlassen oder geschlossen werden kann.

6.1.7 PerfMon

Doppelklicken Sie auf das Piktogramm PerfMon, und das Fenster "Performance Monitor" erscheint. Der obere Teil des Fensters zeigt, in welchem Umfang die Rechenzeit der CPU ausgelastet ist (kalibriert für eine CPU vom Typ 68000). Die schwarze Linie in der Mitte zeigt den Stand bei100%iger Auslastung. Im unteren Teil des Fensters wird gezeigt, wieviel CHIP RAM momentan verwendet wird. Der Umfang des verfügbaren FAST RAM wird am unteren Rand des Bildschirms angegeben.

Im Menü PM kann das Zeitintervall für die Aktualisierung der PerfMon-Anzeige gewählt werden. Die Optionen sind 0,5, 1 und 2 Sekunden. Dem gleichen Zweck dient der rechte Rollbalken.

7. MEMACS

MEMACS (ausgesprochen M-Ihmäks, Abkürzung für MicroEmacs) ist ein bildschirmorientierter Texteditor, der es Ihnen ermöglicht, mehrere Dateien gleichzeitig zu bearbeiten. Die einzige Einschränkung ist, daß der gesamte Umfang jeder Datei gleichzeitig im Speicher Platz finden muß, da MEMACS alle Operationen in speicherresidentem Text ausführt.

Die Länge der Zeilen, die bearbeitet werden können, ist auf 80 Zeichen beschränkt. Zeichen, die sich jenseits des achtzigsten Zeichens in einer Zeile befinden, sind nicht verloren. Sie werden lediglich nicht am Bildschirm angezeigt. Der einzige Weg, diese Zeichen zur Anzeige zu bringen, ist, die Zeile zu teilen oder einige der angezeigten Zeichen zu löschen. Wenn Sie neue Zeichen eingeben, können Sie das auch jenseits des achtzigsten Zeichens. Die dort eingegeben Zeichen werden jedoch nicht am Bildschirm gezeigt.

Sie können MEMACS über den Arbeitstisch oder den Befehlsprozessor aufrufen. Erfolgt der Aufruf über den Arbeitstisch, ist auf das Piktogramm doppelzuklicken. Beim Start über den Befehlsprozessor lautet das Befehlsformat wie folgt:

MEMACS [<Dateiname>] [goto <n>] [OPT W]

[<Dateiname>] Gibt die Datei an, die in MEMACS gelesen werden soll.

[goto <n>] Legt fest, daß der Cursor in Zeile "n" der Datei erscheinen soll.

[OPT W] Öffnet MEMACS in einem Arbeitstisch-Fenster.

7.1 Festlegungen zur Schreibweise und spezielle Begriffe

^(Zeichen) Ein Caret-Zeichen (^) gefolgt von einem Zeichen ist eine "Control-Tastenkombination". Dies bedeutet, daß die Control-Taste gedrückt gehalten werden muß, während die entsprechende Zeichentaste betätigt wird.

'(Zeichen)' Wenn ein Zeichen von Apostrophen eingeschlossen ist, bedeutet das, daß Groß- und Kleinschreibung keine Rolle spielen. Es ist der Buchstabe, der die Funktion wählt. (Diese Schreibweise wird in den Zusammenfassungen am Ende dieses Kapitels häufig benutzt).

<ESC> Stellt die Taste Esc (Escape) auf der Amiga-Tastatur dar.

<DEL> Stellt die Taste Del (Delete) auf der Amiga-Tastatur dar.

<TAB> Stellt die Tabulatortaste auf der Amiga-Tastatur dar.

<RETURN> Stellt die Taste RETURN auf der Amiga-Tastatur dar.

Es gibt im Zusammenhang mit MEMACS einige spezielle Begriffe, die Sie kennen müssen:

Puffer: Ein Speicherbereich, den MEMACS verwaltet. Es wird immer mindestens ein Puffer durch MEMACS verwendet, und er enthält null oder mehr Zeichen des Texts.

Punkt: Die aktuelle Text-Cursorposition.

Marke: Eine Text-Cursorposition, die angegeben werden kann. (Jeder Puffer verfügt über seinen eigenen Punkt und seine eigene Marke.) Der Menübefehl "Set-mark" erlaubt Ihnen, die aktuelle Text-Cursorposition zu "markieren". Sie können den Cursor dann vorwärts und rückwärts durch die Datei bewegen, Text einfügen und löschen. Wenn Sie danach zu der Stelle zurückkehren wollen, die "markiert" wurde, wählen Sie den Befehl "Swap-dot&mark" (Vertausche Punkt und Marke).

Sie können auch eine Marke setzen, um den Anfang eines Textblocks festzulegen, der kopiert, verschoben oder gelöscht werden soll. Der "Block" umfaßt alle Zeichen, die sich zwischen der Marke und der aktuellen Cursorposition befinden.

Kill: Der Befehl "Kill" entfernt Text vom Bildschirm und speichert ihn in einer Zwischenablage (Kill buffer). Dieser Text kann wieder abgerufen und durch den Befehl Yank erneut in das

Dokument eingefügt werden. Wird mehrmals nacheinander der Befehl Kill gewählt (ohne dazwischen den Befehl Yank zu verwenden), wird jeder ausgeschnittene Textblock zum bereits vorhandenen Text in der Zwischenablage **hinzugefügt**.

Yank:

Dieser Befehl kopiert den Inhalt der aktuellen Zwischenablage in die Zeile, die unmittelbar vor der Zeile liegt, in der sich der Cursor befindet. Sie können einen Textblock von einem Puffer in einen anderen kopieren, indem Sie diesen Block durch den Befehl Kill ausschneiden und ihn dann sofort, **ohne den Cursor zuvor zu bewegen,** durch den Befehl Yank an der Ursprungsposition wieder einfügen. Bewegen Sie den Cursor zu einer neuen Position, und wählen Sie erneut "Yank". Der zu kopierende Text ist noch immer in der Zwischenablage enthalten und bleibt dort, bis erneut ein Textblock markiert und durch "Kill" ausgeschnitten wird.

Fenster:

Ein Fenster in MEMACS entspricht nicht einem Fenster auf dem Arbeitstisch. In MEMACS kann der Bildschirm in mehrere Ausschnitte unterteilt werden, so daß Sie mehr als einen Pufferinhalt anzeigen und bearbeiten können bzw. zwei oder mehr unterschiedliche Teile desselben Pufferinhalts. Jeder "Ausschnitt" ist ein MEMACS-Fenster.

Lesen einer Datei:

Wenn Sie MEMACS anweisen, eine Datei zu lesen (Befehl Read-file) wird der Inhalt des aktuellen Puffers durch den Inhalt der gelesenen Datei ersetzt.

Ansprechen einer Datei:

Wenn Sie auf eine neue Datei zugreifen wollen, ohne daß der Inhalt des aktuellen Puffers ersetzt wird, können Sie MEMACS dazu anweisen, die Datei anzusprechen (unter Verwendung des Befehls Visit-file). MEMACS weist dann der angesprochenen Datei einen neuen Puffer zu.

Puffer wählen: Sie können zwischen den einzelnen Puffern, mit denen Sie arbeiten, frei umschalten, indem Sie den Befehl "Select-buffer" wählen und den Namen des Puffers angeben, der verwendet werden soll. MEMACS weist manchmal einem Puffer einen Namen zu, der kürzer ist als der Dateiname, dem er entspricht. Stellen Sie sicher, daß Sie beim Umschalten zwischen den Puffern den korrekten Puffernamen angeben.

Veränderte Puffer: Wenn Sie einen Pufferinhalt ändern, sogar wenn Sie nur die Taste <RETURN> drücken und die Zeilenschaltung dann wieder löschen, hält MEMACS das fest und kennzeichnet den Puffer als veränderten Puffer.

Durch den Befehl "List-buffers" können Sie feststellen, welche Puffer verändert wurden, da diese in der Liste durch einen Stern (*) gekennzeichnet sind. Wenn Sie versuchen, MEMACS zu verlassen, ohne die Änderungen zuvor zu speichern, erscheint eine Eingabeaufforderung. Sie weist Sie darauf hin, daß veränderte Puffer vorhanden sind und fragt, ob Sie das Programm wirklich verlassen wollen. Wird ein Puffer gespeichert, wird das Kennzeichen "verändert" entfernt.

7.2 Öffnen von MEMACS

Wenn Sie MEMACS öffnen, erscheint ein neuer Bildschirm. Am unteren Rand dieses Bildschirms finden Sie die Wörter "MicroEMACS - main". In dieser Zeile wird der Name des momentan verwendeten Puffers angezeigt. In diesem Fall ist das der Hauptpuffer "main". Erinnern Sie sich daran, daß ein Puffer null oder mehr Textzeichen enthalten kann.

Normalerweise werden Sie beim Aufruf von MEMACS einen Dateinamen angeben. Ist die Datei vorhanden, wird sie in den Puffer gelesen. Andernfalls wird die Datei erstellt, wenn Sie Ihre Arbeit speichern.

Sie können mehrere Puffer gleichzeitig benutzen und Sie können mehrere Pufferinhalte gleichzeitig am Bildschirm anzeigen. Menü-Optionen erlauben Ihnen, zwischen diesen umzuschalten. Die Anzeige am Bildschirm entspricht jederzeit dem tatsächlichen Inhalt des Puffers.

Wurde der Inhalt eines Puffers entweder aus einer Datei gelesen oder dorthin geschrieben, wird dieser Puffer dieser Datei zugeordnet. In diesem Fall zeigt die unterste Zeile des Bildschirms den Puffernamen zusammen mit dem Namen der Datei, der er zugeordnet ist.

MEMACS verfügt über zwei Betriebsarten: Normal und Befehl. Wenn sich MEMACS in der Betriebsart Normal befindet, können Sie:

- den Cursor unter Verwendung der Cursor-Steuertasten bewegen;
- den Cursor zum Rand des Fensters bewegen, indem Sie die Taste Shift gedrückt halten und die entsprechende Cursor-Steuertaste betätigen;
- den Cursor durch Klicken mit der linken Maus-Taste auf die gewünschte Position auf dem Bildschirm stellen;
- Zeichen an der aktuellen Cursorposition einfügen, indem Sie diese über die Tastatur eingeben;
- Zeichen an der aktuellen Cursorposition löschen, indem Sie die Taste <DEL> drücken;
- Zeichen links vom Cursor löschen, indem Sie die Taste <BACKSPACE> drücken;
- andere spezielle Funktionen aufrufen, wie im nachstehenden Abschnitt über die Menüs und in der Befehlszusammenfassung am Kapitelende beschrieben.

Wenn sich MEMACS im Befehls-Modus befindet, springt der Cursor in die unterste Zeile des Bildschirms und das Programm fragt Sie nach zusätzlichen Informationen. Der Befehls-Modus wird durch einige Menü-Optionen aktiviert, die später in diesem Kapitel beschrieben sind.

Maus-Befehle

Sie können bei der Arbeit mit MEMACS auch die Maus verwenden. Ist das MEMACS-Fenster inaktiv, wird es aktiviert, indem Sie es mit der Maus anklicken. Sie können auch den Cursor unter Verwendung der Maus bewegen, indem Sie den Maus-Zeiger zu der Position bewegen, an welcher der Cursor erscheinen soll und die linke Maus-Taste drücken. Unter Verwendung der Maus kann auch zwischen Puffern umgeschaltet werden. Dazu müssen Ausgangs- und Zielpuffer jedoch am Bildschirm sichtbar sein.

7.3 MEMACS-Menüs

MEMACS bietet die folgenden Menüs in der Menüzeile:

Project	—	System- und dateiorientierte Funktionen
Edit	—	Befehle für die Dateibearbeitung
Window	—	Steuert die Merkmale der MEMACS-Fenster
Move	—	Steuert die Plazierung des Cursors
Line	—	Zeilenorientierte Operationen
Word	—	Wortorientierte Operationen
Search	—	Optionen für Suchen und Suchen/Ersetzen
Extras	—	Steuert den numerischen Wert von Argumenten und läßt Sie eine Folge von Operationen so durchführen, als würde es sich um einen einzelnen, speziellen Befehl handeln.

In diesem Abschnitt werden alle Menüs und die jeweils darin enthaltenen Befehle erklärt. Jeder der Befehle kann auch über eine Kurzform über die Tastatur gewählt werden. Die Kurzformen erscheinen in den Menüs rechts vom entsprechenden Befehl. In diesem Text werden sie am rechten Rand angegeben.

7.4 Das Menü Project

Die Befehle im Menü Project beeinflussen, mit Ausnahme des Befehls "Visit-file", den Puffer, in dem sich der Cursor momentan befindet.

Rename **^XF**

Verändert den Namen der Datei, die dem aktuellen Puffer zugeordnet ist. Dieser Befehl ist nützlich, wenn im Laufe der Arbeit verschiedene Versionen eines Programms oder einer Textdatei gespeichert werden sollen. Sie können für das Speichern der ersten Version den Befehl Save verwenden, einige Änderungen durchführen, die diesem Puffer zugeordnete Datei umbenennen und dann die neue Version speichern.

Wenn Sie "Rename" wählen, fragt MEMACS:

```
New file name:
```

Wenn Sie nur die Taste <RETURN> drücken, ohne einen Dateinamen anzugeben, wird jede Zuordnung des Puffers zu einer Datei aufgehoben. Soll die

Zuordnung des Puffers zu einer entsprechenden Datei erhalten bleiben, muß ein Dateiname angegeben werden.

Read-file **^X^R**

Ersetzt den Inhalt des aktuellen Puffers durch den Inhalt einer Datei. Wenn Sie den Befehl Read-file wählen, stellt MEMACS den Cursor in die unterste Zeile des Bildschirms und fragt:

```
Read File:
```

Geben Sie den vollständigen Pfad zur Datei ein, inklusive des Datenträgernamens, des Verzeichnisses und des Dateinamens, und drücken Sie <RETURN>. Die Datei wird in den aktuellen Puffer gelesen und überschreibt dabei die Daten, die zuvor dort vorhanden waren.

Soll keine Datei gelesen werden, drücken Sie die Taste <RETURN>, ohne einen Dateinamen einzugeben. MEMACS ignoriert den Befehl und kehrt zum Normal-Modus zurück.

Visit-file **^X^V**

Erlaubt zusätzlich zur ersten geöffneten Datei die Arbeit mit weiteren Dateien. Es muß bereits eine Bearbeitung stattfinden, bevor eine zusätzliche Datei angesprochen werden kann. Dieser Befehl ist nützlich für Programmierer, die bei der Erstellung eines Programms Teile aus einem anderen übernehmen oder sie zu Rate ziehen wollen.

Wenn Sie diesen Befehl aufrufen, stellt MEMACS den Cursor in die unterste Zeile des Bildschirms und fragt:

```
Visit File:
```

Geben Sie den vollständigen Pfad zur Datei ein, und drücken Sie <RETURN>. MEMACS liest die Datei in einen Puffer, wenn sie nicht schon in einem steht. Befindet sich die anzusprechende Datei auf einer anderen Diskette, zeigt AmigaDOS ein Kommunikationsfenster, das Sie auffordert, diese Diskette in ein Laufwerk einzulegen. Befindet sich die Datei bereits in einem Puffer, schaltet MEMACS automatisch zu diesem Puffer um.

Insert-file **^X^I**

Fügt den Inhalt einer Datei in den aktuellen Puffer ein. Wenn Sie diesen Befehl aufrufen, stellt MEMACS den Cursor in die unterste Zeile des Bildschirms und fragt:

```
Insert File:
```

Geben Sie den vollständigen Pfad zur Datei ein, und drücken Sie <RETURN>. MEMACS wird sie im aktuellen Puffer eine Zeile oberhalb der aktuellen Cursorposition einfügen.

Save-file **^X^S**

Schreibt den Inhalt des aktuellen Puffers zu dem Dateinamen, der dem Puffer zugeordnet ist. Der dem Puffer zugeordnete Dateiname wurde bestimmt, als der Inhalt einer existierenden Datei in den Puffer gelesen wurde (**Read-file**) oder als die Datei, die dem aktuellen Puffer zugeordnet ist, umbenannt wurde (**Rename**).

Ist in der Statuszeile kein Dateiname angegeben, meldet MEMACS "No File Name" und bricht den Speicherversuch ab.

Nach der erfolgreichen Speicherung verwendet MEMACS die unterste Zeile des Bildschirms, um Ihnen mitzuteilen, wieviele Zeilen in die festgelegte Datei geschrieben wurden.

Save-file-as **^X^W**

Ermöglicht es, den Namen einer Datei anzugeben, die dem Puffer zugeordnet werden soll. Wenn Sie diesen Befehl aufrufen, fragt MEMACS:

```
Write File:
```

MEMACS fragt nach dem Namen der Datei, in der der Inhalt des aktuellen Puffers gespeichert werden soll. Wenn Sie den vollständigen Pfad zur Datei eingeben und <RETURN> drücken, wird der Pufferinhalt in diese Datei geschrieben. (Wird kein Name angegeben und <RETURN> gedrückt, erfolgt die Rückkehr zum Normal-Modus.) In der Statuszeile des Puffers erscheint folgende Meldung:

```
File: <Dateiname>
```

Von nun an wird diese Datei für die Speicherung des jeweiligen Inhalts dieses Puffers verwendet, wenn der Befehl Save gegeben wird.

Save-mod **^X^M**

Schreibt die Inhalte aller veränderten Puffer zur Diskette. Wenden Sie diesen Befehl mit Vorsicht an, damit nicht unbeabsichtigt ein veränderter Pufferinhalt in die zugeordnete Datei geschrieben wird, obwohl sie nicht verändert werden soll.

Save-exit **^X^F**

Speichert alle veränderten Puffer und beendet das Programm MEMACS. Dieser Befehl ist eine Kombination der Befehle "Save" und "Quit". Wenden Sie auch diesen Befehl mit Vorsicht an (siehe Save-mod).

New-CLI **^-**

Öffnet ein völlig neues CLI-Fenster mit dem Namen "Spawn Window". Durch dieses Fenster können so viele CLI-Befehle wie gewünscht aufgerufen werden, wenn Sie die Arbeit mit MEMACS nicht unterbrechen wollen. Verwenden Sie den Befehl ENDCLI, um zu MEMACS zurückzukehren. Das Fenster verschwindet, und bei MEMACS wird der vorherige Zustand wiederhergestellt.

CLI-Command **^X!**

Erlaubt die Ausführung eines AmigaDOS-Befehls, während Sie sich weiter in MEMACS befinden. Dies ist ähnlich wie beim Aufruf des Befehls RUN, wenn Sie mit dem Befehlsprozessor arbeiten. Wird dieser Menü-Befehl gewählt, stellt MEMACS den Cursor in die unterste Bildschirmzeile und fordert Sie zur Eingabe auf (!). Sie können dann in dieser Zeile einen Befehl eingeben, den AmigaDOS verarbeiten soll. MEMACS unterbricht vorübergehend den Betrieb, und AmigaDOS führt Ihren Befehl aus. Die Ausgabe des Befehl erscheint in einem temporären Puffer mit dem Namen "spawn.output".

Quit **^C**

Beendet MEMACS. Wurde seit der letzten Speicherung einer oder mehrere der Pufferinhalte verändert, fragt MEMACS:

```
Modified buffers exist, do you really want
to exit? [y/n]
(Puffer wurden modifiziert, wollen Sie wirklich beenden?)
```

MEMACS gibt Ihnen dadurch die letzte Möglichkeit, Ihre Arbeit zu speichern. Wenn Sie das Programm nicht beenden wollen, drücken Sie <RETURN>. Wenn Sie MEMACS verlassen wollen, geben Sie "y" ein und drücken Sie dann <RETURN>.

Vor der Beendigung können Sie prüfen, auf welche Puffer MEMACS sich bezieht, indem Sie den Befehl "List-buffers" aus dem Menü "Edit" wählen. Es werden die Namen aufgelistet, die jedem Puffer zugeordnet sind. Puffer, die nach dem zuletzt vorgenommenen Speichervorgang erneut verändert wurden, sind mit einem Stern (*) gekennzeichnet.

Es gibt Umstände, unter denen die Pufferinhalte nicht zu den Originaldateien zurückgespeichert werden sollen. Nehmen Sie z. B. an, daß Sie ein Programm schreiben und dazu Stücke aus anderen, bereits existierenden Programmen kopieren. Einige der durch "Visit-file" angesprochenen Dateien könnten dabei versehentlich verändert werden oder sich auf einer schreibgeschützten Diskette befinden.

Wenn Sie ein altes Programm vorübergehend als Quellenmaterial für ein neues verwenden wollen, soll es meistens nicht zerstört werden. Deshalb sollten Sie das neue Programm nach der Erstellung speichern und MEMACS verlassen, ohne den veränderten Puffer zur alten Programmdatei zurückzuschreiben.

Zwei weitere Tastaturbefehle für den Aufruf des Befehls "Quit" sind ^X^C und <ESC>^C.

7.5 Das Menü Edit

Die Befehle im Menü Edit beziehen sich auf die Bearbeitung Ihrer Puffer und deren zugeordnete Dateien.

Kill-region ^W

Schneidet einen Textblock aus dem aktuellen Puffer aus und speichert ihn in einer Zwischenablage (Kill buffer). Dies ist ein spezieller Puffer für Text, der unter Verwendung des Befehls Kill aus Puffern gelöscht wurde. (Der Text kann durch den nachstehend beschriebenen Befehl Yank wieder in das Dokument eingefügt werden.)

Sie können den Befehl Kill-region auch verwenden, um einen Block von einem Abschnitt des Puffers in einen anderen zu kopieren. Markieren Sie diesen Block und schneiden Sie ihn durch den Befehl Kill aus. Fügen Sie ihn dann sofort, ohne den Cursor zuvor zu bewegen, durch Yank wieder an der Ursprungsposition ein. Der Block erscheint an der Originalposition, es ist jedoch auch eine Kopie in der Zwischenablage enthalten.

Wenn Sie mehrmals nacheinander den Befehl Kill-region für verschiedene Textbereiche anwenden, ohne dazwischen den Befehl Yank zu wählen, werden die einzelnen ausgeschnittenen Bereiche in der Zwischenablage aufeinanderfolgend gespeichert. Durch den ersten Aufruf des Befehls Yank wird das Ende der Zwischenablage markiert.

Yank ^Y

Kopiert den Inhalt der Zwischenablage in die Zeile des aktuellen Puffers, die sich unmittelbar oberhalb der aktuellen Cursorposition befindet. Der Befehl Yank stellt die Umkehrung des Befehls Kill-region dar, er verändert jedoch den Inhalt der Zwischenablage nicht. Deshalb kann der Cursor mehrmals zu anderen Puffern bewegt und der Inhalt der Zwischenablage dorthin kopiert werden. Der alte Inhalt der Zwischenablage wird jedoch ersetzt und geht verloren, sobald durch den Befehl Kill-region ein neuer Textblock ausgeschnitten wird.

Die Befehle Kill-region und Yank werden häufig zusammen verwendet, um Text von einem Puffer an einen anderen zu übergeben.

Set-mark **^@**

Markiert die Cursorposition in einem Puffer. Durch das Wählen des Befehls Set-mark wird die Position des Cursors im aktuellen Puffer **markiert**. Ab diesem Zeitpunkt wird jede andere Position des Cursors als Punkt bezeichnet. Zwischen der Marke und einem Punkt kann durch den Befehl Swap-dot&mark im Menü Move beliebig hin- und hergesprungen werden.

Sie können den Befehl Set-mark verwenden, um den Anfang eines Textblocks zu markieren, der kopiert oder zu einer anderen Position im Puffer verschoben werden soll. Setzen Sie die Marke am ersten Zeichen, das der Block beinhalten soll. Bewegen Sie dann den Cursor durch die Datei. Der Text zwischen der Marke und der jeweiligen Cursorposition wird als Block betrachtet.

Ein weiterer Tastaturbefehl für Set-mark ist **<ESC>-**.

Copy-region **<ESC>w**

Kopiert den Inhalt des markierten Bereichs zur Zwischenablage. Dieser neue Text ersetzt den vorherigen Inhalt der Zwischenablage.

Upper-region **^X^U**

Ändert den Text des gesamten markierten Bereichs zwischen der Marke und der aktuellen Cursorposition (Punkt) zu Großbuchstaben.

Lower-region **^X^L**

Ändert den Text des gesamten markierten Bereichs zu Kleinbuchstaben.

List-buffers **^X^B**

Teilt das Fenster des aktuellen Puffers und gibt eine Liste der Puffer aus, die durch MEMACS momentan gehalten werden. Die Liste besteht aus vier Spalten. Zum Beispiel:

```
CSize    Buffer File
*17260   emacs.doc       df1:docfiles/emacs.doc
```

C ist eine Abkürzung für "verändert" (changed) und zeigt bei Puffern, die seit der letzten Speicherung verändert wurden, einen Stern.

Size zeigt, wieviele Zeichen sich im jeweiligen Puffer befinden.

Buffer gibt den Namen des jeweiligen Puffers an. Wurde eine Datei gelesen (read-file), wird er normalerweise den Namen dieser Datei, ohne Pfad, tragen. Ist der Name der bearbeiteten Datei z. B. df1:docfiles/emacs.doc, so trägt der Puffer den Namen emacs.doc.

File gibt den Namen der Datei, inklusive des vollständigen Pfads, an. Dies zeigt Ihnen, wohin MEMACS die Datei schreibt, wenn einer der Befehle Save oder Save-exit gewählt wird, während der Cursor sich in diesem Puffer befindet.

Wird der Befehl List-buffers gewählt, erscheint in der Statuszeile am unteren Rand des Bildschirms "MEMACS - [List]". Obwohl durch List-buffers ein Fenster erscheint, wird es nicht als verfügbarer Puffer aufgelistet. Wenn Sie im Fenster von List-buffers arbeiten wollen, können Sie dies wie bei jedem anderen Puffer tun. Wird z. B. eine Datei geöffnet und in dieses Fenster gelesen, wird der Name des Puffers weiterhin [List] bleiben und der Name der geöffneten Datei wird dem Fenster von List-buffers zugewiesen.

Sie können das Fenster von List-buffers am Bildschirm behalten, während Sie ein anderes Fenster benutzen, um die aufgelisteten Puffer zu bearbeiten. Die Anzeige im Fenster von List-buffers wird jedoch nicht aktualisiert, um die Änderungen anderer Puffer anzugeben. Für das Abrufen der aktuellen Informationen muß der Befehl List-buffers erneut gewählt werden.

Select-buffer **^Xb**

Erlaubt die Auswahl, welcher Puffer im **momentan gewählten Fenster** (das Fenster, in dem sich der Cursor befindet) bearbeitet werden soll. Wenn Sie den Befehl Select-buffers wählen, stellt MEMACS den Cursor in die unterste Zeile und fragt:

```
Use buffer:
```

Es muß ein Name eingegeben werden, der identisch mit einem aus der Liste von List-buffers ist. Wird einer der verfügbaren Namen angegeben, ersetzt der Inhalt dieses Puffers den Inhalt des momentan gewählten Fensters.

Wird ein Namen angegeben, der nicht in der Liste von List-buffers enthalten ist, faßt MEMACS dies als Anweisung auf, einen neuen Puffer mit diesem Namen zu erstellen. In diesem Fall ist dem neuen Puffer kein Dateiname zugeordnet und

die Datei muß umbenannt oder der Befehl Save-as-file gewählt werden, wenn der Inhalt dieses Puffers in einer Datei gespeichert werden soll.

Wird nur <RETURN> gedrückt, ohne einen Namen einzugeben, wird der Befehl ignoriert.

Insert-buffer **<ESC>^Y**

Fügt den Inhalt des angegebenen Puffers eine Zeile oberhalb der aktuellen Cursorposition in den aktuellen Puffer ein. Wird der Befehl Insert-buffer gewählt, fragt MEMACS:

```
Insert buffer:
```

Der Name des einzufügenden Puffers muß angegeben und dann die Taste <RETURN> gedrückt werden.

Kill-buffer **^Xk**

Löscht den Inhalt des angegebenen Puffers. MEMACS kann eine Datei nur dann bearbeiten, wenn der gesamte Dateiinhalt im verfügbaren Speicher Platz findet. Sie können den Befehl Kill-buffer verwenden, um den Inhalt eines oder mehrerer Puffer zu löschen und so im Systemspeicher Platz zu schaffen. Durch diesen Befehl wird der zuvor für einen Puffer benutzte Speicher für die erneute Verwendung an die Speicherverwaltung zurückgegeben.

Wird der Befehl Kill-buffers gewählt, fragt MEMACS:

```
Kill buffer:
```

Nun muß der Name des zu löschenen Puffers eingegeben werden. Ein Puffer, dessen Inhalt momentan angezeigt wird, kann nicht gelöscht werden.

Justify-buffer **^XJ**

Entfernt alle leeren Bereiche und Tabulatoren an den Zeilenanfängen im aktuellen Puffer. Der Text wird so umgestaltet, daß er entlang der aktuellen Ränder ausgerichtet ist.

Redisplay **^L**

Bewirkt, daß der gesamte Bildschirminhalt erneut dargestellt wird.

Quote-char **^Q**

Erlaubt das Einfügen von Steuerzeichen als Textzeichen in eine Datei. Einige Tastatureingaben wurden MEMACS als Steuerzeichen zugewiesen (z. B. die Befehlskurzformen). Wenn Sie versuchen, eine davon in Ihren Text einzufügen, reagiert MEMACS, als wäre eine Menüoption gewählt worden.

Zum Beispiel wird MEMACS durch Control-L (^L) angewiesen, die Anzeige erneut darzustellen. ^L ist jedoch auch als Steuerung beim Drucken nützlich, um ein Zeichen für Seitenvorschub einzufügen. Durch das Wählen von Quote-char faßt MEMACS das nächste eingegebene Zeichen immer als Textzeichen auf und fügt es in die Datei ein, anstatt es als Menübefehl zu behandeln.

Soll eine Control-Tastenkombination angegeben werden, ist ein Caret (^) gefolgt von einem Zeichen zu verwenden, ebenso wie in diesem Handbuch beschrieben. Von MEMACS wird diese Kombination von Caret und Zeichen jedoch als ein Zeichen betrachtet. Dies gilt für die Cursor-Steuertasten ebenso wie für die Zeichenzählung.

Sie können den Befehl Quote-char auch verwenden, um das Zeichen für die Betätigung der Taste <RETURN> in den Text einzufügen, ebenso wie jedes andere Steuerzeichen, das während eines Makrobefehls erforderlich ist. Sogar ^Q kann eingefügt werden, indem es zweimal eingegeben wird.

Ein weiterer Tastaturbefehl für Quote-char ist **^Xq**.

Indent **^J**

Bewegt den Cursor zur nächsten Zeile und rückt diese automatisch um dieselbe Anzahl von Leerzeichen wie die vorherige Zeile ein.

Transpose **^T**

Tauscht die Positionen von zwei aufeinanderfolgenden Zeichen. Plazieren Sie den Cursor auf das rechte der beiden Zeichen.

Cancel **^G**

Beendet einen laufenden Menübefehl, wie z. B. den Befehl Query-s-r (Suchen und Ersetzen mit Abfrage).

7.6 Das Menü Window

Ein Fenster in MEMACS unterscheidet sich von einem Fenster im Anzeigesystem des Arbeitstischs. MEMACS teilt den Bildschirm in mehrere Ausschnitte und ermöglicht das Bearbeiten separater Dateien (Puffer) in jedem MEMACS-Fenster. Das Menü Window erlaubt Ihnen zu steuern, wie die Pufferinhalte am Bildschirm dargestellt werden.

One-window **^X1**

Stellt den Inhalt des aktuellen Puffers in einem einzigen Fenster der vollen Größe im MEMACS-Fenster dar. Alle anderen Puffer bleiben unsichtbar, wodurch Ihnen der maximale Platz für die Bearbeitung des aktuellen Puffers zur Verfügung steht.

Split-window **^X2**

Teilt das aktuelle Fenster in zwei Hälften, wobei der aktuelle Pufferinhalt in beiden Fenstern identisch dargestellt wird. Dies erlaubt Ihnen die Bearbeitung von zwei Teilen des Puffers gleichzeitig. Alle Änderungen, die in einem der beiden Fenster durchgeführt werden, beeinflussen den gesamten Puffer und damit auch das andere Fenster. Diese Option ist nützlich, wenn Sie bei der Arbeit an einem hinteren Teil Ihres Dokuments einen weit davor liegenden Teil sehen wollen.

Next-window **^Xn**

Bewegt den Cursor zum nächsten Fenster "nach unten" und stellt dieses für die Bearbeitung zur Verfügung.

Prev-window **^Xp**

Bewegt den Cursor zum nächsten Fenster "nach oben" und stellt dieses für die Bearbeitung zur Verfügung.

Die Befehle "Next-window" und "Prev-Window" verfügen über keine Sperre. Wurde der Cursor zum untersten Fenster bewegt, springt er automatisch zum obersten, wenn erneut der Befehl Next-window gegeben wird. Umgekehrt springt der Cursor auch zum untersten Fenster, falls er sich im obersten befindet, wenn Sie den Befehl Prev-window wählen.

Expand-window **^Xz**

Vergrößert das aktuelle Fenster um eine Zeile und verkleinert das nachfolgende dabei automatisch um eine Zeile.

Shrink-window **^X^Z**

Verkleinert das aktuelle Fenster um eine Zeile und vergrößert das nachfolgende dabei automatisch um eine Zeile.

Next-w-page **<ESC>^V**

Zeigt die nächste Seite im nächsten Fenster an. Wurde z. B. ein Fenster geteilt, und Sie arbeiten im oberen von beiden, bewirkt der Befehl Next-w-page, daß der Inhalt des unteren Fensters (in dem Sie nicht arbeiten) um eine Seite weitergeblättert wird. Dadurch wird das Fenster nicht für die Bearbeitung aktiviert; der Inhalt kann lediglich eingesehen werden.

Prev-w-page **^Xv**

Zeigt die nächste Seite des vorhergehenden Fensters. Wird nur ein Fenster angezeigt, wird der Inhalt in diesem um eine Seite weitergeblättert.

7.7 Das Menü Move

Die Befehle im Menü Move ermöglichen die schnelle Bewegung des Cursors durch den gesamten Puffer.

Top-of-buffer **<ESC><**

Bewegt den Cursor zur obersten Zeile des aktuellen Puffers.

End-of-buffer **<ESC>>**

Bewegt den Cursor zur untersten Zeile des aktuellen Puffers.

Top-of-window **<ESC>,**

Bewegt den Cursor zur obersten Zeile des aktuellen Fensters.

End-of-window **<ESC>.**

Bewegt den Cursor zur untersten Zeile des aktuellen Fensters.

Goto-line **^X^G**

Bewegt den Cursor zur Zeile mit der angegebenen Nummer. Wenn Sie den Befehl Goto-line wählen, stellt MEMACS den Cursor auf den unteren Rand des Bildschirms und fragt:

```
goto-line:
```

Geben Sie eine Zeilennummer ein, drücken Sie <RETURN> und MEMACS bewegt den Cursor direkt zu dieser Zeile. Wird eine Zeilennummer angegeben, die die Gesamtanzahl der Zeilen im Puffer überschreitet, wird der Cursor zur letzten Zeile des Puffers bewegt.

Swap-dot&mark **^X^X**

Setzt eine Marke an der aktuellen Cursorposition und bewegt den Cursor zu der Position, an der eine Marke gesetzt wurde. Wurde in diesem Fenster noch keine Marke gesetzt, antwortet MEMACS mit "No mark in this window". Dieser Befehl erlaubt die schnelle Bewegung des Cursor zu einer vorher festgelegten Position im Puffer. Wird er erneut gewählt, kehrt der Cursor an die Position zurück, an der er sich befand, als der Befehl Swap-dot&mark zuvor gewählt wurde.

Next-page **^V**

Blättert den Text im Fenster um ein komplettes Fenster, abzüglich einer Zeile, weiter nach vorne. Die Position des Cursors wird verändert, so daß er weiter am Bildschirm angezeigt wird.

Prev-page **<ESC>v**

Blättert den Text im Fenster um ein komplettes Fenster, abzüglich einer Zeile, zurück. Die Position des Cursor wird verändert, so daß er weiter am Bildschirm angezeigt wird.

Next-word **<ESC>f**

Bewegt den Cursor vorwärts (f wie "forward") zum nächsten nicht alphabetischen Zeichen nach dem aktuellen Wort.

Previous-word **<ESC>b**

Bewegt den Cursor rückwärts (b wie "before") zum ersten nicht alphabetischen Zeichen vor dem aktuellen Wort.

Scroll-up **^Z**

Rollt den Text im Fenster um eine Zeile weiter.

Scroll-down **<ESC>z**

Rollt den Text im Fenster um eine Zeile zurück.

7.8 Das Menü Line

Die Befehle im Menü Line ermöglichen die Cursorbewegung innerhalb einer oder zwischen Zeilen und erlauben die Durchführung von Operationen, die eine Zeile ganz oder teilweise betreffen.

Open-line **^O**

Teilt die Zeile, in der sich der Cursor befindet, wobei das Zeichen an der Cursorposition als erstes Zeichen in der folgenden Zeile erscheint. Dieser Befehl beläßt den Cursor in der Originalzeile, so daß, beginnend an der aktuellen Cursorposition, neue Zeichen geschrieben werden können.

Wird der Befehl Open-line versehentlich gewählt, kann durch sofortiges Drücken der Taste <DEL> die Zeile wieder zusammengefügt werden.

Kill-line **^X^D**

Löscht die Zeile, in der sich der Cursor befindet und überträgt den Text in die Zwischenablage. Wurde seit dem letzten Kill-Befehl nicht der Befehl Yank gewählt, wird die Zeile an den in der Zwischenablage vorhandenen Text **angefügt**.

Kill-to-eol **^K**

Schneidet den Text von der aktuellen Cursorposition bis zum Zeilenende aus und überträgt ihn in die Zwischenablage. Wurde seit dem letzten Kill-Befehl nicht der Befehl Yank gewählt, wird die Zeile an den in der Zwischenablage vorhandenen Text **angefügt**. Er kann wieder in der Datei eingefügt werden, indem sofort der Befehl Yank gewählt wird.

Start-of-line **^A**

Bewegt den Cursor zur äußersten Position links in einer Zeile (A wie Anfang).

End-of-line **^E**

Bewegt den Cursor zur äußersten Position rechts in einer Zeile (E wie Ende). Wurden mehr als 80 Zeichen in einer Zeile geschrieben, erscheint ein Dollarzeichen ($) am Zeilenende. Das Bewegen des Cursors zum Zeilenende bewirkt, daß er logisch zum äußersten Zeichen rechts gesetzt wird, obwohl dieses nicht zu sehen ist. Physisch wird der Cursor beim Dollarzeichen positioniert. Wenn Sie die Cursor-Steuertaste mit dem Pfeil nach links betätigen, um den Cursor nach links zu bewegen, kann es erforderlich sein, sie so oft zu drücken, wie nicht sichtbare Zeichen vorhanden sind, bevor sich der Cursor tatsächlich bewegt.

Next-line **^N**

Bewegt den Cursor um eine Zeile abwärts.

Previous-line **^P**

Bewegt den Cursor um eine Zeile aufwärts.

Line-to-top **<ESC>!**

Bewegt die Zeile, die den Cursor enthält, zum oberen Rand des Fensters.

Delete-blanks **^X^O**

Löscht ausgehend von der Cursorposition in Richtung auf das Pufferende alle Leerzeilen, bis MEMACS die nächste Zeile findet, die Text enthält.

Show-Line# **^X=**

Zeigt Informationen über die aktuelle Cursorposition. Zum Beispiel:

```
Line 17 Column 1 (2%)
```

In diesem Beispiel befindet sich der Cursor in der siebzehnten Zeile des Texts in der ersten Spalte. Die Prozentangabe zeigt, daß der Bereich zwischen der Cursorposition und dem oberen Rand des Puffers 2% des Gesamt-Pufferumfangs darstellt. Mit anderen Worten, würde sich der Cursor beim letzten Zeichen des Textes befinden, würde die Prozentzahl mit 100 angegeben werden.

7.9 Das Menü Word

Das Menü Word enthält Befehle, die sich auf einzelne Wörter auswirken.

delete-forw **<ESC>d**

Löscht das Zeichen an der Cursorposition und alle übrigen, folgenden Zeichen des Wortes bis zum nächsten, nicht alphabetischen Zeichen (d. h. einem Leerschritt, Tabulator oder Satzzeichen).

Wird z. B. der Cursor beim "b" des Wortes "Menübefehl" plaziert und delete-forw gewählt, wird der Teil "befehl" des Wortes gelöscht. Befindet sich der Cursor bei einem Leerzeichen, muß er zum Beginn des Wortes vorwärts bewegt werden, bevor dieses gelöscht werden kann.

delete-back **<ESC>h**

Löscht das Zeichen an der Cursorposition und alle übrigen, vorhergehenden Zeichen des Wortes bis zum Wortanfang.

Eine weitere Tastenkombination für diesen Befehl ist **<ESC><DEL>.**

Upper-word **<ESC>u**

Ändert die Zeichen eines Wortes zu Großbuchstaben, beginnend mit dem Zeichen an der Cursorposition bis zum Wortende.

Lower-word **<ESC>l**

Ändert die Zeichen eines Wortes zu Kleinbuchstaben, beginnend mit dem Zeichen an der Cursorposition bis zum Wortende.

Cap-word **<ESC>c**

Ändert das Zeichen eines Wortes, an dem sich der Cursor befindet, zu einem Großbuchstaben. Der Befehl ändert außerdem alle weiteren Zeichen des Wortes bis zum Wortende zu Kleinbuchstaben.

Switch-case **<ESC>^**

Tauscht die Groß-/Kleinschreibung eines Wortes beginnend mit der Cursorposition bis zum Wortende. Ist das Wort in Großbuchstaben geschrieben, wird es zu Kleinbuchstaben geändert und umgekehrt.

7.10 Das Menü Search

Das Menü Search erlaubt Ihnen, im aktuellen Puffer nach speziellen Zeichenfolgen zu suchen. Die Groß- oder Kleinschreibung wird bei der Suche nicht berücksichtigt. Wenn Sie jedoch die Funktion Suchen und Ersetzen verwenden, wird die Groß-/Kleinschreibung des Ersatzbegriffs in den Text übernommen.

Search-forward **^S**

Sucht ausgehend von der aktuellen Cursorposition in Richtung auf das Pufferende. Wird dieser Befehl gewählt, stellt MEMACS den Cursor auf die unterste Zeile des Bildschirms und fragt:

```
Search:
```

Geben Sie die Zeichenfolge ein, nach der MEMACS suchen soll, und drücken Sie <RETURN>. Wird die Zeichenfolge gefunden, positioniert MEMACS den Cursor unmittelbar hinter dem letzten Zeichen der Zeichenfolge.

Kann MEMACS die Zeichenfolge nicht finden, erscheint die Meldung "Not found".

Eine weitere Tastenkombination für diesen Befehl ist **^Xs**.

Search-backward **^R**

Sucht ausgehend von der aktuellen Cursorposition rückwärts in Richtung auf den Pufferanfang. Dieser Befehl arbeitet ansonsten ebenso wie der Befehl Search-forward. Eine alternative Form dieses Befehls ist **^Xr**.

Search-replace **<ESC>r**

Arbeitet wie der Befehl Search-forward, mit Ausnahme der Tatsache, daß die Eingabe einer Ersatz-Zeichenfolge mit unterschiedlichem Text möglich ist. Wenn MEMACS das erste Auftreten des angegebenen Suchbegriffs findet, erscheint die Frage:

```
Replace:
```

Sie müssen eine Folge von Zeichen eingeben, die die gefundene Zeichenfolge ersetzen soll. Beachten Sie, daß die Groß-/Kleinschreibung der Zeichen so im Text erscheint, wie sie eingegeben wurde. Sobald Sie die Taste <RETURN> drücken, sucht MEMACS bis zum Pufferende weiter nach der angegebenen Zeichenfolge und ersetzt sie jedesmal durch die Ersatz-Zeichenfolge. Nach Beendigung der Ausführung meldet MEMACS:

```
Replaced (xx) occurences.
```

(xx) gibt an, wie oft die gesuchte Zeichenfolge ersetzt wurde.

Query-s-r **<ESC>q**

Arbeitet wie der Befehl Search-replace, erlaubt jedoch im Einzelfall zu entscheiden, ob das Ersetzen stattfinden soll. Wird Query-s-r gewählt, fragt MEMACS zunächst nach der zu suchenden Zeichenfolge, dann nach dem Ersatztext:

```
Query replace:
```

Wird eine übereinstimmende Zeichenfolge gefunden, erscheint immer die Abfrage:

```
Change string? [y/n/c/^G]?
```

Die Optionen lauten: y für "Ja", n für "Nein", c für "an allen weiterhin gefundenen Positionen automatisch ersetzen" und ^G für "abbrechen". Nach Beendigung der Befehlsausführung meldet MEMACS:

```
Replaced (xx) occurences
```

Fence-match **<ESC>^F**

Findet das nächste Auftreten des Zeichens an der aktuellen Cursorposition. Befindet sich der Cursor z. B. bei einem Stern (*), wird er zu der Position bewegt, an der der nächste Stern im Text vorhanden ist.

7.11 Das Menü Extras

Das Menü Extras enthält Befehle, durch die Sie die Arbeitsweise von MEMACS festlegen können. Viele dieser Befehle erfordern, daß numerische Argumente angegeben werden, bevor der eigentliche Befehl gewählt wird. Dieses Menü beinhaltet auch mehrere Makrobefehle. Ein Makrobefehl besteht aus einer Folge von Einzelbefehlen oder Tastenbetätigungen, die ausgeführt werden, wenn aus dem Menü **Execute-macro** gewählt wird.

Set-arg **^U**

Für das Festlegen eines numerischen Arguments für einen Operationsbefehl. MEMACS bewegt den Cursor zur untersten Zeile des Bildschirms und fragt:

```
Arg: 4
```

Wenn Sie Set-arg erneut wählen, wird der Wert des Arguments mit 4 multipliziert.

Wenn Sie eine der numerischen Tasten (0 bis 9, NICHT auf dem Ziffernblock der Tastatur!) drücken, akzeptiert MEMACS ein ganzzahliges Argument. Wird zuerst das Minus-Symbol eingegeben, akzeptiert MEMACS negative, ganzzahlige Argumente, beginnend mit -1.

Beispiele: (jedes durch die Betätigung von ^U eingeleitet)

Arg: -1	("-" als erste Taste gedrückt)
Arg: -23	("- 2 3" als Dreiertastenfolge)
Arg: 12	("3 ^U" als Zweiertastenfolge)

MEMACS akzeptiert den Argumentenwert als Schlüssel für die nächste auszuführende Aktion. Sollen 12 Leerzeilen an der aktuellen Cursorposition eingegeben werden, geben Sie das Argument mit 12 an und drücken die Taste <RETURN>.

Sollen 20 Minuszeichen eingefügt werden, legen Sie als Argument 20 fest, **drücken nicht die Taste <RETURN>**, sondern das Minus-Symbol auf der Tastatur. (Hinweis: Verwenden Sie nicht das Minus-Symbol des numerischen Tastenfelds. Es ist mit einem anderen Wert adressiert.)

Soll einer der Operationsparameter von MEMACS (nachstehend beschrieben) gesetzt werden, legen Sie den Wert des Arguments fest und **drücken nicht die Taste <RETURN>**, sondern wählen Sie den entsprechenden Menübefehl. MEMACS verwendet dann das Argument als Wert für den Befehl.

Set **<ESC>s**

Erlaubt, verschiedene MEMACS-Parameter zu wählen. Wird Set gewählt, antwortet MEMACS mit:

```
Set:
```

Sie können einen der folgenden Begriffe eingeben:

Screen	Plaziert die MEMACS-Anzeige in ein Fenster des Arbeitstischs oder wieder in einen Benutzerbildschirm.
Interlace	Schaltet die Zwischenzeilenabtastung ein und aus.
Mode	bewirkt als zweite Eingabeaufforderung die Meldung "Mode:". Es kann **cmode** (für die Bearbeitung von C-Programmen) oder **wrap*** (für die Aktivierung des automatischen Wortumbruchs, sobald beim Schreiben von Text eine festgelegte Cursorposition erreicht wird) eingegeben werden. Mit Cmode kann automatisch ein mit dem Zeichen an der Cursorposition identisches Zeichen gesucht werden. Verwenden Sie **+mode** oder **-mode**, um einen Modus hinzuzufügen oder zu entfernen.
Left*	Bestimmt den linken Rand
Right*	Bestimmt den rechten Rand
Tab*	Setzt den Wert für die Abstände der Tabulatoren
Indent*	Wird im Cmode verwendet, um festzulegen, wie weit jede Stufe der Verschachtelung eingerückt werden soll.
Case	Schaltet die Berücksichtigung der Groß-/Kleinschreibung beim Suchen ein oder aus; Standardeinstellung ist "aus".
Backup	Schaltet die Sicherungsfunktion von MEMACS ein oder aus. Die Optionen lauten ON (benennt die aktuelle Datei beim Speichern in <Dateiname>.bak um und speichert diese Sicherungsdatei im Verzeichnis T:), SAFE (diese Option prüft, ob bereits eine Datei für den Puffer vorhanden ist. Wenn ja, wird die existierende Datei nicht überschrieben.) und OFF (Standardoption, MEMACS führt keine Sicherung durch).

* Jeder dieser Begriffe bewirkt eine Eingabeaufforderung für ein numerisches Argument, sofern dieses nicht bereits zusammen mit dem Begriff angegeben wurde.

Start-macro **^X(**

Weist MEMACS an, alle nachfolgenden Tastenbetätigungen oder Menü-Wahlvorgänge aufzuzeichnen. Dies ist ein Makrobefehl und wird in Verbindung mit den Befehlen Stop-macro und Execute-macro verwendet.

Stop-macro **^X)**

Weist MEMACS an, die Aufzeichnung der Tastenbetätigungen zu beenden.

Execute-macro **^Xe**

Wiederholt die Tastenbetätigungen und Menü-Wahlvorgänge, die zwischen den Befehlen Start-macro und Stop-macro vorgenommen wurden. Sie werden wiederholt, als wäre die gesamte Befehlsfolge neu eingegeben worden.

Set-key **^X^K**

Durch diesen Befehl wird Ihnen ermöglicht, allen Funktionstasten, Funktionstasten in Verbindung mit Shift, der Hilfetaste und allen Tasten des numerischen Tastenfelds Tastaturmakros zuzuweisen. Wenn Sie dann eine dieser neu definierten Tasten bei der Makroaufzeichnung betätigen, wird die neue Tastendefinition im Befehl aufgezeichnet. Für jede dieser Tasten kann eine Definition festgelegt werden, die bis zu 80 Tastenanschläge enthalten darf.

HINWEIS: Wenn Sie den Befehl Set-mark in die Definition eines Tastaturmakros aufnehmen wollen, darf nicht die Tastaturkurzform ^@ verwendet werden, da sie nicht korrekt arbeitet. Sie müssen statt dessen die alternative Form für Set-mark, "<ESC>-" benutzen. Diese Form wird von Tastaturmakros akzeptiert.

Wenn Sie Set-key wählen, fragt MEMACS:

```
key to define:
```

Drücken Sie eine der 10 Funktionstasten, die Hilfetaste oder eine Taste des numerischen Tastenfelds. MEMACS antwortet mit:

```
def: [Befehle]:
```

[Befehle] ist die Anzeige der Befehle, die momentan an die Taste gebunden sind. Geben Sie eine neue Folge von Zeichen (bis zu 80) ein, auf die MEMACS

reagieren soll, wenn diese Taste gedrückt wird. Das Drücken der Taste <RETURN> beendet die Eintragung.

Beachten Sie, daß für die Eingabe von Befehlen, die Funktionstasten darstellen, wie z. B. <ESC>< (Cursor zum Pufferanfang), der Befehl Quote-char [^Q] verwendet werden muß. Dadurch werden die Funktionen nicht aufgerufen, sondern als Zeichen in die Definition eingefügt.

Die nachstehende Tabelle zeigt die Standardwerte der Funktionstasten, wenn sie in Makrobefehlen verwendet werden:

TASTE	STANDARDWERT	TASTENFOLGE
Funktionstaste 1	Zeile vervielfachen	^A^K^Y^M^Y
Funktionstaste 2	Zeile löschen	^X^D
Funktionstaste 3	Tastaturmakro ausführen	^Xe
Funktionstaste 4	Nächster Bildschirm	^V
Funktionstaste 5	Vorheriger Bildschirm	<ESC>v
Funktionstaste 6	Fenster teilen	^X2
Funktionstaste 7	Ein Fenster	^X1
Funktionstaste 8	Fenster nach oben blättern	^Z
Funktionstaste 9	Fenster nach unten blättern	<ESC>Z
Funktionstaste 10	Datei sichern und MEMACS beenden	^X^F
Hilfetaste	Zeile einfügen und einrücken	^J
Enter im numerischen Tastenfeld	Zeile einfügen und einrücken	^J

Die numerischen Tasten, der Punkt und die Minus-Taste im numerischen Tastenfeld besitzen standardgemäß ihre normalen Funktionen (d. h. Standardwert der Taste 1 ist 1, der Taste 2 ist 2, etc.)

Reset-keys **<ESC>k**

Weist allen durch den Befehl Set-keys neu definierten Tasten wieder den Standardwert zu.

Execute-file **<ESC>e**

Erlaubt während der Arbeit mit MEMACS die Ausführung einer Programmdatei. Wird der Befehl gewählt, antwortet MEMACS mit:

```
File:
```

Geben Sie den Namen der Datei ein, auf die Sie zugreifen wollen. Sie wird als Datei von MEMACS-Befehlen ausgeführt.

Execute-line **^[^[**

Schaltet MEMACS zum Befehls-Modus um. Wird Execute-line gewählt, fragt MEMACS:

```
execute-line:
```

Sie können nun den Namen jedes Menübefehls und seine Parameter über die Tastatur eingeben. Es muß exakt die Schreibweise verwendet werden, inklusive der Bindestriche, die im Menü angegeben ist, da sonst ein Alarm ertönt und die Meldung "command error" erscheint. Es kann z. B. nicht eingegeben werden:

```
execute-line: insert file <Dateiname>
```

Sie müssen schreiben:

```
execute-line: insert-file <Dateiname>
```

Eine alternative Tastenkombination für execute-line ist **<ESC><ESC>.**

7.12 Befehle, die nicht in den Menüs vorhanden sind

Die folgenden Befehle sind nicht in den Menüs vorhanden und können nur über die Tastatur eingegeben werden.

Taste beschreiben **<ESC>^D**

Teilt Ihnen mit, ob Funktionen an eine Taste oder Tastenkombination gebunden sind. Wenn Sie <ESC>^D eingeben, fragt MEMACS nach der zu beschreibenden Taste. Wird eine Tastenfolge eingegeben, wie z. B. ^L oder <ESC>k, antwortet MEMACS mit der Bezeichnung der entsprechenden Funktion. In diesem Beispiel also mit "Redisplay" bzw. "Reset-keys".

Binden einer Funktion an eine Taste **<ESC>^B**

Erlaubt Ihnen, einer Taste eine Funktion zuzuweisen. Wenn MEMACS nach der Taste fragt, geben Sie die Funktion (entsprechend der Schreibweise, die im Menü verwendet wird) ein, und drücken Sie dann die Taste oder die Tastenfolge. Für die Prüfung, ob die Funktion der Taste korrekt zugewiesen wurde, kann der Befehl Taste beschreiben (<ESC>^D) verwendet werden.

Lösen einer Funktion von einer Taste **<ESC>^U**

Ermöglicht Ihnen, die Bindung einer Taste oder einer Tastenfolge an eine Funktion zu lösen. Wenn MEMACS nach der Taste fragt, deren Bindung gelöst werden soll, drücken Sie die Taste oder die Tastenfolge. MEMACS antwortet mit "Key is not bound".

Echo **<ESC>^E**

Zeigt die eingegebene Zeichenfolge in der Befehlszeile an. Dieser Befehl wird normalerweise verwendet, wenn zur Verwendung unter MEMACS ausführbare Programmdateien erstellt oder bearbeitet werden.

Zur Fensterkante bewegen **<Shift> + Cursor-Steuertaste**

Durch das Gedrückthalten der Taste Shift und die Betätigung einer Cursor-Steuertaste bewegt sich der Cursor zur oberen, unteren, linken oder rechten Kante des Bildschirms. Dies ist abhängig vom Umfang des verfügbaren Texts.

Nächstes Zeichen löschen **^D**

Löscht das Zeichen an der aktuellen Cursorposition. Die Wirkung dieses Befehls ist identisch mit der der Taste <DEL>.

Vorheriges Zeichen löschen **^H**

Löscht das Zeichen links von der aktuellen Cursorposition. Die Wirkung dieses Befehls ist identisch mit der der Taste <BACKSPACE>.

Zur nächsten Zeile bewegen **^M**

Fügt nach der aktuellen Cursorposition ein Zeichen für "neue Zeile" ein und bewegt den Cursor zum Beginn der neuen Zeile.

Cursor um x Zeichen bewegen **(vorwärts)** **^F**
(rückwärts) **^B**

Erlaubt Ihnen, den Cursor um eine angegebene Anzahl von Zeichen in Vorwärts- oder Rückwärtsrichtung zu bewegen. Der Standardwert für diesen Befehl ist 1. Es kann jedoch unter Verwendung von ^U ein höherer Argumentenwert angegeben werden. Geben Sie danach ^F oder ^B ein, um den Cursor um diese Anzahl von Zeichen zu bewegen.

7.13 Hinzufügen von Befehlen zur MEMACS-Startdatei

Wird MEMACS geöffnet, liest es den Inhalt einer Datei mit dem Namen emacs_pro, um festzustellen, ob Befehle vorhanden sind, die automatisch ausgeführt werden sollen. Emacs_pro ist nicht vorhanden, sondern muß von Ihnen erstellt werden. MEMACS sucht zuerst im aktuellen Verzeichnis nach emacs_pro. Ist die Datei dort nicht vorhanden, wird die Suche im Verzeichnis s:emacs der Diskette fortgesetzt.

Falls gewünscht, können Sie mehr als eine Datei emacs_pro erstellen. Es kann eine allgemeine Datei angelegt werden, die bei jedem Öffnen von MEMACS bewirkt, daß eine Folge von Befehlen ausgeführt wird. Und Sie können spezifische Dateien mit Startbefehlen erstellen, die sich auf eine häufig verwendete Datei beziehen. Sind spezifische Dateien und eine allgemeine Datei mit dem Namen emacs_pro vorhanden, setzen die spezifischen Start-Befehlsdateien die allgemeine Datei außer Kraft.

7.14 Nach Funktion sortierte Zusammenfassung der Befehle

Operationen:

Pufferinhalt durch neue Datei ersetzen (Read-file) ..... ^X^R
Zusätzliche Datei öffnen (Visit-file) ..... ^X^V
Datei in aktuellen Puffer einfügen (Insert-file) ..... ^X^I
Puffer umbenennen ..... ^XF
Liste der Puffer anzeigen ..... ^X^B
Puffer zur Bearbeitung wählen ..... ^Xb
Pufferinhalt in aktuellen Puffer einfügen ..... <ESC>^Y
Pufferinhalt in neue Datei schreiben (Save-as-file) ..... ^X^W
Puffer in existierende Datei speichern ..... ^X^S
Alle veränderten Puffer speichern ..... ^X^M
Alle veränderten Puffer speichern und beenden ..... ^X^F
Numerischen Argumentenwert setzen ..... ^U
MEMACS-Parameter setzen ..... <ESC>s
Aktuelle Cursorposition zeigen (Show-line#) ..... ^X=
Aufruf eines CLI-Befehls ..... ^X!
Öffnen eines CLI-Fensters ..... ^-
Menübefehl abbrechen ..... ^G
MEMACS beenden ..... ^C
^X^C
<ESC>^C

Cursorbewegung:

Zwischen Punkt und Markenposition wechseln ..... ^X^X
x Zeichen vorwärts bewegen ..... ^F*
x Zeichen rückwärts bewegen ..... ^B*
Zum Zeilenanfang ..... ^A
Zum Zeilenende ..... ^E
Zur angegebenen Zeile ..... ^X^G
Zur nächsten Zeile ..... ^N
Zur vorherigen Zeile ..... ^P
Zur nächsten Seite ..... ^V
Zur vorherigen Seite ..... <ESC>v
Zum Pufferanfang ..... <ESC><
Zum Pufferende ..... <ESC>>
Zum nächsten Wort ..... <ESC>f
Zum vorherigen Wort ..... <ESC>b

*Wird in Verbindung mit Set-arg (^U) verwendet.

Fensterbefehle:

Fenster des aktuellen Puffers in voller Größe (ein Fenster) ..... ^X1
Aktuelles Fenster teilen ..... ^X2
Fenster vergrößern ..... ^Xz
Fenster verkleinern ..... ^X^Z
Cursor zum oberen Fensteranfang ..... <ESC>,
Cursor zum unteren Fensterende ..... <ESC>.
Aktuelle Zeile zum oberen Fensterrand ..... <ESC>!
Fensterinhalt um eine Zeile nach oben rollen ..... ^Z
Fensterinhalt um eine Zeile nach unten rollen ..... <ESC>z
Cursor zum nächsten Fenster bewegen ..... ^Xn
Cursor zum vorherigen Fenster bewegen ..... ^Xp
Bildschirm neu darstellen ..... ^L

Text:

Marke im Text setzen ..... ^@
<ESC>-
Zeichen als Textzeichen betrachten ..... ^Q
^Xq
Text im Puffer ausrichten ..... ^XJ
Zeichenpositionen tauschen ..... ^T
Wort in Großbuchstaben darstellen ..... <ESC>u
Wort in Kleinbuchstaben darstellen ..... <ESC>l
Bereich in Großbuchstaben darstellen ..... ^X^U
Bereich in Kleinbuchstaben darstellen ..... ^X^L
Groß-/Kleinschreibung eines Wortes tauschen ..... <ESC>^
Nur erstes Zeichen des Wortes groß schreiben ..... <ESC>c
Tabulator einfügen ..... ^I
<TAB>
Zeile einfügen, Cursor zur nächsten Zeile bewegen ..... ^M
<RETURN>
Zeile einfügen, Cursor zur nächsten Zeile bewegen und Zeile um ebensoviele Leerschritte wie die vorherige einrücken ..... ^J
Zeile teilen (Open-line) ..... ^O

Löschen und Kopieren:

Nächstes Zeichen löschen ..... ^D
<DEL>
Vorheriges Zeichen löschen ..... ^H
<BACKSPACE>
Nächstes Wort löschen ..... <ESC>d
Vorheriges Wort löschen ..... <ESC>h
<ESC><DEL>
Text von Cursor bis Zeilenende löschen ..... ^K
Gesamte Zeile löschen ..... ^X^D
Bereich zwischen Punkt und Marke löschen ..... ^W
Leerzeilen löschen ..... ^X^O
Gesamten Puffer löschen ..... ^Xk
Inhalt des markierten Bereichs in Zwischenablage kopieren ..... <ESC>w
Inhalt der Zwischenablage in aktuellen Puffer kopieren (YANK) ..... ^Y

Suchen:

Vorwärts suchen ..... ^S
^Xs
Rückwärts suchen ..... ^R
^Xr
Vorwärts suchen und ersetzen ..... <ESC>r
Vorwärts suchen, abfragen und ersetzen ..... <ESC>q
Identisches Zeichen suchen ..... <ESC>^F

Makrobefehle:

Anfang eines Makrobefehls ..... ^X(
Ende eines Makrobefehls ..... ^X)
Ausführung eines Makrobefehls ..... ^Xe
<ESC>~
Zeile ausführen ..... ^[^[
<ESC><ESC>
Datei ausführen ..... <ESC>e
Funktionstasten definieren (Set-key) ..... ^X^K
Tasten zurücksetzen ..... <ESC>k
Taste beschreiben ..... <ESC>^D
Taste binden ..... <ESC>^B
Bindung einer Taste lösen ..... <ESC>^U
Echo-Zeichenfolge ..... <ESC>^E

7.15 Alphabetisch sortierte Zusammenfassung der Befehle

Zur Beachtung: Wird eine Taste zwischen Apostrophen angegeben, bedeutet dies, daß die Groß-/Kleinschreibung des Zeichens keine Rolle spielt. Es ist das Zeichen, das auf der Tastenkappe angegeben ist, das die Funktion auslöst.

Control-Tastenkombinationen

Marke setzen ..... ^'@'
Neuer Befehlsprozessor CLI ..... ^'-'
Zeile ausführen ..... ^[^[
Cursor zum Zeilenanfang ..... ^'A'
x Zeichen rückwärts bewegen ..... ^'B'*
MEMACS beenden ..... ^'C'
Nächstes Zeichen löschen ..... ^'D'
Zum Zeilenende ..... ^'E'
x Zeichen vorwärts bewegen ..... ^'F'*
Menübefehl abbrechen ..... ^'G'
Vorheriges Zeichen löschen ..... ^'H'
Tabulator einfügen ..... ^'I'
Zeile einfügen, Cursor zur nächsten Zeile bewegen und Zeile
um ebensoviele Leerschritte wie die vorherige einrücken ..... ^'J'
Text bis Zeilenende ausschneiden ..... ^'K'
Bildschirm neu darstellen ..... ^'L'
Zeile einfügen, Cursor zur nächsten Zeile bewegen ..... ^'M'
Zur nächsten Zeile ..... ^'N'
Zeile teilen (Open-line) ..... ^'O'
Zur vorherigen Zeile ..... ^'P'
Steuerzeichen als Textzeichen betrachten ..... ^'Q'
Rückwärts suchen ..... ^'R'
Vorwärts suchen ..... ^'S'
Zeichenpositionen tauschen ..... ^'T'
Numerischen Argumentenwert setzen ..... ^'U'
Zur nächsten Seite ..... ^'V'
Bereich zwischen Punkt und Marke löschen ..... ^'W'
Text aus Zwischenablage einfügen (Yank) ..... ^'Y'
Fensterinhalt nach oben rollen ..... ^'Z'

* Wird in Verbindung mit dem Befehl Set-arg (^U) verwendet.

Control-X/Control-Tastenkombinationen

Liste der Puffer anzeigen ..... ^X^B
MEMACS beenden ..... ^X^C
Zeile ausschneiden ..... ^X^D
Alle veränderten Puffer speichern und beenden ..... ^X^F
Cursor zur angegebenen Zeile ..... ^X^G
Datei in aktuellen Puffer einfügen (Insert-file) ..... ^X^I
Funktionstasten definieren (Set-key) ..... ^X^K
Bereich in Kleinbuchstaben darstellen ..... ^X^L
Alle veränderten Puffer speichern ..... ^X^M
Leerzeilen löschen ..... ^X^O
Pufferinhalt durch neue Datei ersetzen (Read-file) ..... ^X^R
Puffer in existierende Datei speichern ..... ^X^S
Bereich in Großbuchstaben darstellen ..... ^X^U
Zusätzliche Datei öffnen (Visit-file) ..... ^X^V
Pufferinhalt in neue Datei schreiben (Save-as-file) ..... ^X^W
Zwischen Punkt und Markenposition wechseln ..... ^X^X
Fenster verkleinern ..... ^X^Z

Control-X/Tastenkombination

Ein Befehl des Befehlsprozessors CLI ..... ^X!
Anfang eines Makrobefehls (Start-macro) ..... ^X(
Ende eines Makrobefehls (Stop-macro) ..... ^X)
Aktuelle Cursorposition zeigen (Show-line#) ..... ^X=
Fenster des aktuellen Puffers in voller Größe (ein Fenster) ..... ^X1
Aktuelles Fenster teilen ..... ^X2
Puffer zur Bearbeitung wählen ..... ^X'b'
Ausführung eines Makrobefehls ..... ^X'e'
Puffer umbenennen ..... ^X'F'
Text im Puffer ausrichten ..... ^X'J'
Gesamten Puffer ausschneiden ..... ^X'k'
Cursor zum nächsten Fenster bewegen ..... ^X'n'
Cursor zum vorherigen Fenster bewegen ..... ^X'p'
Steuerzeichen als Textzeichen betrachten ..... ^X'q'
Suche nach rückwärts ..... ^X'r'
Suche nach vorwärts ..... ^X's'
Nächste Seite des vorherigen Fensters ..... ^X'v'
Fenster vergrößern ..... ^X'z'

Escape-Tastenkombinationen

Aktuelle Zeile zum oberen Fensterrand ..... <ESC>!
Groß-/Kleinschreibung eines Wortes tauschen ..... <ESC>^
Marke setzen ..... <ESC>-
Cursor zum oberen Fensteranfang ..... <ESC>,
Cursor zum unteren Fensterende ..... <ESC>.
Cursor zum Pufferanfang ..... <ESC><
Cursor zum Pufferende ..... <ESC>>
Taste binden ..... <ESC>^B
Cursor zum vorherigen Wort ..... <ESC>b
MEMACS beenden ..... <ESC>^C
Nur erstes Zeichen des Wortes groß schreiben ..... <ESC>c
Taste beschreiben ..... <ESC>^D
Nächstes Wort löschen ..... <ESC>d
Echo-Zeichenfolge ..... <ESC>^E
Datei ausführen ..... <ESC>e
Identisches Zeichen suchen ..... <ESC>^F
Zum nächsten Wort ..... <ESC>f
Vorheriges Wort löschen ..... <ESC>h
Tasten zurücksetzen ..... <ESC>k
Wort in Kleinbuchstaben darstellen ..... <ESC>l
Vorwärts suchen, abfragen und ersetzen ..... <ESC>q
Vorwärts suchen und ersetzen ..... <ESC>r
MEMACS-Parameter setzen ..... <ESC>s
Bindung einer Taste lösen ..... <ESC>^U
Wort in Großbuchstaben darstellen ..... <ESC>u
Nächste Seite des nächsten Fensters anzeigen ..... <ESC>^V
Zur vorherigen Seite ..... <ESC>v
Inhalt des markierten Bereichs in Zwischenablage kopieren ..... <ESC>w
Pufferinhalt in aktuellen Puffer einfügen ..... <ESC>^Y
Fensterinhalt um eine Zeile nach unten rollen ..... <ESC>z
Zeile ausführen ..... <ESC><ESC>
Vorheriges Wort löschen ..... <ESC><DEL>

Funktionstasten

Nächstes Zeichen löschen ..... <DEL>
Vorheriges Zeichen löschen ..... <BACKSPACE>
Zeile einfügen, Cursor zur nächsten Zeile bewegen ..... <RETURN>
Tabulator einfügen ..... <TAB>

A. Für Amiga 2000 mit Festplattenlaufwerk...

Wenn Sie einen Amiga 2000 mit einem Festplattenlaufwerk besitzen, bzw. planen eines zu installieren, gibt es in der Version 1.3 einige Details, die speziell für Sie bedeutsam sind. Erstens können Sie nun, wenn Sie eine Festplatten-/SCSI-Steuerung vom Typ A2090A besitzen, den Systemstart automatisch von der Festplatte ausführen. Außerdem gibt es ein neues Dateisystem, FastFileSystem (FFS), das die Geschwindigkeit beim Zugriff und bei der Verwendung von Dateien auf der Festplatte erhöht.

A.1 Die Steuerplatine A2090A

Die A2090A ist die neue Festplatten-/SCSI-Steuerplatine für den Amiga 2000. Sobald das Kickstart-ROM der Version 1.3 in Ihrem Amiga 2000 installiert ist, können Sie den Systemstart automatisch von der Festplatte ausführen. Es ist dann nicht erforderlich, eine Workbench-Diskette oder eine für den Neustart geeignete Diskette eines Anwendungsprogramms in das Diskettenlaufwerk einzulegen! Die A2090A wird durch die Zusammenarbeit zwischen dem Kickstart-ROM der Version 1.3 und den beiden Autoboot-ROMs auf der A2090A in die Lage versetzt, den Neustart automatisch von der Platte durchzuführen. (Ihre A2090A ist u. U. noch nicht mit diesen ROMs ausgestattet. Im Lieferumfang dieser Platine sind jedoch vollständige Installationsanweisungen für das Einsetzen dieser Bausteine enthalten.)

A.2 Das schnelle Dateisystem (FastFileSystem)

Das FastFileSystem (FFS) ist ein neues Dateisystem für die Verwendung mit allen Festplattenlaufwerken, die direkt (nicht über Janus und PC) an den Amiga angeschlossen werden können. Die einzige Einschränkung ist, daß der Festplattentreiber den Standardsatz an Plattenbefehlen bereitstellen und das Medium im Format von Sektoren und Spuren anbieten muß. Da die Festplatten-/SCSI-Steuerung von Commodore das Standard-Dateisystem automatisch in der ersten Partition einer Festplatte anmeldet, ist es nicht möglich, das FastFileSystem dort zu benutzen. Jedoch können bei Verwendung des Befehls MOUNT in der neuen Version 1.3 weitere Partitionen angelegt werden, die mit dem FastFileSystem arbeiten.

Da die meisten Festplattenlaufwerke für die Datenübertragung die Methode DMA (direct memory access - direkten Speicherzugriff) verwenden, ist das alte Dateisystem ineffizient, weil es nicht die volle Leistung der Hardware ausnützt. FFS beseitigt dieses Problem, indem in den Datenblöcken ausschließlich Daten und weniger Organisationsinformationen gespeichert werden. Wann immer es möglich ist, werden die Datenblöcke unmittelbar aufeinanderfolgend angeordnet, so daß umfassende Lese- und Schreibvorgänge in einer Operation durchgeführt werden können. Ein angenehmer Nebeneffekt ist, daß durch das neue Datenformat 4,9 % mehr Daten auf einer Festplatte gespeichert werden können als bisher. Das sind etwa 50 KByte je MByte Plattenspeicher.

Es ist etwas schwierig, die Erhöhung der Geschwindigkeit in Zahlen zu fassen. FFS kann unfragmentierte Dateien mit der Geschwindigkeit lesen, mit der das Festplattenlaufwerk sie übertragen kann. Das Schreiben von Dateien erfolgt mit ca. 75 % der maximalen Übertragungsgeschwindigkeit der Platte. Durch FFS kann eine Festplatte mit normaler Kapazität (etwa 20 MByte) ca. vierzigmal schneller validiert werden. FFS hält Schritt mit Erhöhungen der Plattenübertragungsgeschwindigkeiten und nützt die Datenbandbreite einer vorhandenen Steuerung voll aus. Das Dateisystem ist mit den meisten existierenden Software-Produkten kompatibel, mit Ausnahme einiger Programme vom Type "DiskDoctor", die nur das alte Datenblockformat kennen. Die neue Programmversion "DiskDoctor" auf der Workbench-Diskette der Version 1.3 kann für das normale Dateisystem und das FastFileSystem eingesetzt werden.

Anmelden des FFS

Die Treibersoftware A2090(A) meldet die erste Festplattenpartition automatisch als Gerät an, das mit dem normalen Dateisystem arbeitet. Das FastFileSystem kann nur in den zusätzlichen angelegten Partitionen verwendet werden. Da die Partitionen unter Verwendung des Befehls MOUNT angemeldet werden, kann in den Einträgen in der Datei MountList ein anderes Dateisystem festgelegt werden. Diese Partitionen müssen jedoch unter Verwendung von FFS im FFS-Format (erneut) formatiert werden.

Ziel kann sein, eine relativ kleine Partition im herkömmlichen Dateisystem mit nur den allernötigsten Daten und Programmen für den Neustart einzurichten und den großen Rest der Platte unter FFS zu betreiben.

1. Sichern Sie zuerst den Inhalt des Festplattenlaufwerks, oder lagern Sie ihn aus.
2. Führen Sie unter Verwendung der Workbench-Diskette der Version 1.3 einen Neustart durch, oder kopieren Sie vor dem Neustart die neuen Dateien l:FastFileSystem, c:Mount und c:Format auf Ihre Startdiskette.

3a. FALLS AUF DER FESTPLATTE BEREITS DIE GEWÜNSCHTEN PARTITIONEN ANGELEGT SIND:

Wenn auf der Festplatte bereits Partitionen angelegt sind und die Größe der ersten als Partition mit dem normalen Dateisystem akzeptabel ist, ist es wahrscheinlich möglich, das erneute Partitionieren mit PREP und Formatieren der gesamten Festplatte zu umgehen. (Die erste Partition sollte normalerweise relativ klein sein.) Ist das der Fall, muß für jede zusätzliche Partition der Eintrag in der Datei DEVS/MountList um drei Zeilen erweitert werden:

```
GlobVec         =1
FileSystem      = L:FastFileSystem
DosType         = 0x444F5301
```

(**ACHTUNG**: Die Werte von "LowCyl" und HighCyl", welche die Größe der Partition bestimmen, dürfen nicht geändert werden! Werden diese Werte geändert, muß die gesamte Platte neu mit PREP bearbeitet und formatiert werden - siehe 3b. Wenn Sie die Namen Ihrer Partitionen ändern, müssen die in der Startsequenz angegebenen Namen ebenfalls entsprechend geändert werden.)

Führen Sie nun einen Neustart des Computers durch. Wenn Ihre Start-Befehlsdatei diese Partitionen anmeldet, erscheinen Kommunikationsfenster mit der Meldung "Not A DOS Disk". Brechen Sie diese durch Cancel ab, und formatieren Sie dann die FFS-Partitionen mit dem neuen FORMAT-Befehl. Trägt Ihre Partition z. B. den Namen fs1:

```
FORMAT DRIVE fs1: NAME "FAST2" FFS
```

ACHTUNG: ALLE IN DIESER PARTITION ENTHALTENEN DATEN GEHEN VERLOREN! STELLEN SIE VOR DEM FORMATIEREN SICHER, DASS KOPIEN DER DATEIEN VORHANDEN SIND.

3b. WENN AUF DER FESTPLATTE DIE PARTITIONEN NOCH NICHT WUNSCHGEMÄSS VORHANDEN SIND:

Sind auf der Festplatte die Partitionen noch nicht wunschgemäß angelegt, muß die gesamte Platte neu mit PREP bearbeitet und formatiert werden. Ändern Sie die Datei MountList und geben Sie Einträge für eine oder mehrere FFS-Partitionen ein. Die Einträge von LowCyl und HighCyl sind davon abhängig, wieviele Zylinder während des Programms PREP für die Anfangspartition angegeben werden sollen, die automatisch angemeldet wird und mit dem normalen Dateisystem arbeitet.

Einige Benutzer ziehen eine kleine Anfangspartition (Zylinder 2 und 3) vor, während andere eine Kapazität von mindestens 880 KByte wünschen, um den Inhalt der Workbench-Diskette dort unterzubringen. Die folgenden Gleichungen sollen Ihnen helfen zu bestimmen, wieviele Zylinder erforderlich sind, um den Inhalt der Workbench-Diskette aufzunehmen:

Bytes je Zylinder	=	Blöcke je Spur x Oberflächen x 512
Bytes je Diskette	=	880 x 1.024 = 901.120
Anzahl der Zylinder für Aufnahme einer Diskette	=	Bytes je Diskette / Bytes je Zylinder (Ergebnis aufrunden)

Bei vielen Festplattenlaufwerken mit ST506-Schnittstelle und 20 MByte sind z. B. 26 Zylinder erforderlich.

Verwenden Sie PREP wie im A2090-Handbuch beschrieben, und legen Sie Ihre Partitionen an.

Unser Beispiellaufwerk wird so präpariert, daß die Zylinder 0 bis 1 reserviert werden (RES0:), und die Zylinder 2 bis 28 die erste Paritition darstellen (wird als DH0: automatisch angemeldet und arbeitet mit dem normalen Dateisystem). Die verbleibenden Zylinder 29 bis 611 stehen für Partitionen zur Verfügung, die mit dem FastFileSystem arbeiten.

Nach Beendigung des Vorbereitungsprogramms PREP müssen ein Neustart und dann die folgenden Schritte durchgeführt werden. Im hier aufgeführten Beispiel wird angenommen, daß mit einem Festplattenlaufwerk vom Type ST506 mit 20 MByte Kapazität gearbeitet wird und daß für die erste Partition die Zylinder 2 bis 28 belegt werden. **Hinweis: Wird ein anderes Laufwerk verwendet, müssen die Laufwerksparameter entsprechend geändert werden.**

1. Rufen Sie die Datei DEVS/MountList zur Bearbeitung auf und fügen Sie die nachstehende Beschreibung Ihrer FFS-Partition ein. Die Zeilen mit GlobVec, FileSystem und DosType sind bei allen FFS-Einträgen erforderlich. Zum Beispiel:

```
FS1:  Device = hddisk.device
      Unit = 1
      Flags = 0
      Surfaces = 4
      BlocksPerTrack = 17
      Reserved = 2
      Interleave = 0
      LowCyl = 29
      HighCyl = 611
      Buffers = 20
      Stacksize = 4000
      GlobVec = 1
      FileSystem = L:FastFileSystem
      DosType = 0x444F5301
```

2. Formatieren Sie die erste Partition (dh0:) für das normale Dateisystem:

```
FORMAT DRIVE DH0: NAME AmigaHD
```

3. Melden Sie Ihre FFS-Partition an:

```
MOUNT FS1:
```

4. Formatieren Sie die FFS-Partition für das FastFileSystem:

```
FORMAT DRIVE FS1: NAME "FAST1" FFS
```

Die Option FFS gibt an, daß die Festplatte für die Verwendung des FastFileSystem formatiert werden soll. Wird kein Schlüsselwort angegeben, prüft der Befehl FORMAT den Eintrag DosType in der Datei MountList, um festzustellen, ob dort ein

anderes Dateisystem als das normale festgelegt ist. Ist "FastFileSystem" eingetragen, wird die Option FFS automatisch verwendet. Die Optionen FFS und NOFFS können benutzt werden, um diesen Prozeß zu unterbinden.

5. Bearbeiten Sie Ihre Startsequenz, und fügen Sie nach der Zeile BINDDRIVERS die Zeile "MOUNT FS1:" ein, um die Partition beim Start anzumelden.

Verwendung der resetfesten RAM-Disk mit FFS

Sie können für die resetfeste RAM-Disk das FastFileSystem einsetzen, dann kann jedoch von dort ausgehend kein Neustart durchgeführt werden. Soll die resetfeste RAM-Disk mit dem FFS arbeiten, muß in der Datei MountList der Wert für BootPri im Eintrag für die resetfeste RAM-Disk (RAD:) zu -129 geändert werden. Dieser Wert zeigt dem System, daß ausgehend von der resetfesten RAM-Disk kein Neustart erfolgen kann.

B. Liste der Verzeichnisse/Dateien auf der Workbench-Diskette

Verzeichnis C

ADDBUFFERS	Anweisung an das Dateisystem, zusätzliche Cache-Puffer anzulegen.
ASK	Fragt beim Ausführen einer Befehlsdatei nach einer Benutzereingabe.
ASSIGN	Weist einem Verzeichnis einen logischen Gerätenamen zu.
AVAIL	Gibt Informationen über den verfügbaren CHIP-RAM- und FAST-RAM-Speicher aus.
BINDDRIVERS	Verbindet Gerätetreiber mit der Hardware.
BREAK	Setzt Unterbrechungskennungen im angegebenen Prozeß.
CD	Setzt, ändert oder meldet das aktuelle Verzeichnis.
CHANGETASKPRI	Ändert die Priorität von Prozessen, die über den Befehlsprozessor gestartet werden.
COPY	Kopiert eine Datei oder ein Verzeichnis.
DATE	Anzeigen oder Setzen des Systemdatums und /oder der Systemzeit.
DELETE	Löscht bis zu 10 Dateien oder Verzeichnisse.
DIR	Anzeige einer sortierten Liste des Inhalts eines Verzeichnisses.
DISKCHANGE	Informiert den Amiga darüber, daß in einem 5-1/4-Zoll-Laufwerk eine Diskette gewechselt wurde.
DISKDOCTOR	Versuch, eine fehlerhafte Diskette zu reparieren.
ECHO	Druckt eine Zeichenfolge.
ED	Ruft Textdatei zur Bearbeitung auf (ein Bildschirm-Editor).
EDIT	Ruft Textdatei zur zeilenweisen Bearbeitung des Quelltexts auf (ein Zeileneditor).
ELSE	Erlaubt Angabe einer Alternative bei Bedingungen in einer Befehlsdatei.

ENDCLI	Beendigung eines interaktiven CLI- oder SHELL-Prozesses.
ENDIF	Beendet einen IF-Block in einer Befehlsdatei.
ENDSKIP	Beendet einen SKIP-Block in einer Befehlsdatei.
EVAL	Errechnet Ergebnisse einfacher Ausdrücke.
EXECUTE	Führt Befehlsdateien unter Angabe von Argumenten aus.
FAILAT	Setzt die Fehlerbedingung in einer Befehlsdatei.
FAULT	Druckt die angegebene(n) Fehlermeldung(en).
FF	Beschleunigt die Anzeige von Text.
FILENOTE	Weist einer Datei einen Kommentar zu.
GETENV	Ruft den Wert einer Umgebungsvariablen ab.
ICONX	Ruft eine AmigaDOS-Befehlsdatei über den Arbeitstisch auf.
IF	Steuert Bedingungsoperationen in einer Befehlsdatei.
INFO	Liefert Informationen über das Dateisystem.
INSTALL	Schreibt bzw. überprüft den Boot-Block auf einer formatierten Diskette.
JOIN	Verbindet bis zu 15 Dateien zu einer neuen Datei.
LAB	Gibt eine Sprungadresse in einer Befehlsdatei an.
LIST	Listet die angeforderten Informationen über Verzeichnisse und Dateien auf.
LOADWB	Startet den Arbeitstisch.
LOCK	Stellt den Status einer Festplatte, die mit FFS arbeitet, auf "schreibgeschützt" ein.
MAKEDIR	Erstellt ein neues Verzeichnis.
MOUNT	Meldet ein neues AmigaDOS-Gerät an.
NEWCLI	Startet einen neuen interaktiven CLI-Prozeß in einem neuen Fenster.
NEWSHELL	Startet einen neuen interaktiven SHELL-Prozeß in einem neuen Fenster.
PATH	Legt den Pfad fest, den der Befehlsprozessor für die Suche nach Befehlen verwendet.
PROMPT	Ändert die Zeichenfolge der Eingabeaufforderung der aktuellen SHELL.
PROTECT	Ändert die Schutzbits einer Datei.
QUIT	Verläßt eine Befehlsdatei mit einem angegebenen Fehlercode.
RELABEL	Ändert den Datenträgernamen einer Diskette.

REMRAD	Entfernt die resetfeste RAM-Disk.
RENAME	Ändert den Namen einer Datei oder eines Verzeichnisses.
RESIDENT	Lädt Befehle resident und fügt sie der Liste der residenten Befehle hinzu.
RUN	Läßt Befehle als Hintergrundprozesse ablaufen.
SEARCH	Sucht in Datei(en) nach der angegebenen Textzeichenfolge.
SETCLOCK	Setzt oder liest die Einstellung der Echtzeituhr.
SETDATE	Ändert den Datums- und Zeiteintrag einer Datei oder eines Verzeichnisses.
SETENV	Setzt den Wert einer Umgebungsvariablen.
SETPATCH	Korrigiert Fehler der Kickstart-Versionen 1.2 und 1.3.
SKIP	Sprungbefehl bei der Ausführung von Befehlsdateien.
SORT	Sortiert kleine Dateien.
STACK	Zeigt oder setzt die Größe des Stack-Speichers für den aktuellen Befehlsprozessor.
STATUS	Listet die Informationen über die CLI-/SHELL-Prozesse auf.
TYPE	Gibt den Inhalt einer Textdatei aus.
VERSION	Findet die Versions- und Revisionsnummer des Arbeitstischs, von Bibliotheken oder Geräten.
WAIT	Bewirkt, daß AmigaDOS für die angegebene Zeit in den Wartezustand versetzt wird.
WHICH	Sucht nach dem Befehlspfad für den angegebenen Befehl.
WHY	Druckt eine Fehlermeldung, die den Grund für das Fehlschlagen eines Befehls angibt.

Verzeichnis System

CLI	Erlaubt die direkte Kommunikation mit dem Amiga über AmigaDOS-Befehle.
DiskCopy	Kopiert den Inhalt einer Diskette auf eine andere.
FastMemFirst	Ändert die Zuordnungsreihenfolge der Speicher; setzt FAST-RAM an erste Stelle.
FixFonts	Aktualisiert die Dateien .fonts im Verzeichnis FONTS:

FORMAT	Initialisiert eine Diskette für die Verwendung mit dem Amiga.
InitPrinter	Initialisiert einen Drucker für die Verwendung mit dem Amiga.
MergeMem	Mischt die Speicherlisten von separat konfigurierten RAM-Platinen.
NoFastMem	Zwingt den Amiga, ausschließlich mit dem residenten CHIP-RAM zu arbeiten.
SetMap	Setzt die Tastenbelegung für die Tastatur.

Verzeichnis Utitlities

CALCULATOR	Standardrechner mit vier Funktionen.
Clock	Zeigt eine Digital- oder Analoguhr auf dem Arbeitstisch.
ClockPtr	Ändert den Mauspfeil in eine Digitaluhr.
CMD	Leitet die serielle oder parallele Ausgabe auf eine Datei um.
GraphicDump	Erstellt einen Grafikausdruck des vordersten Bildschirms.
InstallPrinter	Kopiert einen Druckertreiber von der Extras- zur Workbench-Diskette.
MORE	Zeigt den Inhalt von ASCII-Textdateien an.
Notepad	Erlaubt die Erstellung kurzer Textdateien.
PrintFiles	Kopiert mehrere Dateien zum Drucker.
Say	Sprachausgabe-Programm.

Verzeichnis Prefs

CopyPrefs	Kopiert die Datei System-configuration (mit den Preferences-Einstellwerten) von einer Festplatte zur Workbench-Diskette.
Preferences	Erlaubt die Änderung verschiedener Einstellungen des Amiga, inklusive der Bildschirmfarben, des verwendeten Druckertreibers, der Form und Farbe des Zeigers und der Einstellungen derr seriellen Schnittstelle.

Verzeichnis S

CLI-Startup	Die Datei, die bei jedem neuen CLI-Start ausgeführt wird.
DPAT	Eine Befehlsdatei, durch die bei Befehlen mit zwei Argumenten mit Jokerzeichen gearbeitet werden kann.
PCD	Eine Befehlsdatei, ähnlich dem Befehl CD, die sich an das Ausgangsverzeichnis "erinnert".
SHELL-Startup	Die Datei, die bei jedem neuen SHELL-Start ausgeführt wird.
SPAT	Eine Befehlsdatei, durch die bei Befehlen mit genau einem Argument mit Jokerzeichen gearbeitet werden kann.
Startup-Sequence	Die Befehlsdatei, die bei jedem Neustart der Workbench-Diskette ausgeführt wird.
Startup-Sequence.hd	Eine Startsequenz, die die Steuerung an eine Festplatte übergibt.
StartupII	Eine Befehlsdatei, die durch die Datei Startup-Sequence gestartet wird und viele AmigaDOS-Befehle resident lädt.

Verzeichnis Fonts

Zeichensätze	*Schriftgrößen*
Ruby	8, 12, 15
Opal	9, 12
Sapphire	14, 19
Diamond	12, 20
Garnet	9, 16
Emerald	17, 20
Topaz	11

Verzeichnis Devs

clipboard.device	Steuert das Schreiben und Lesen von Zwischenspeicherinhalten zu oder von CLIPS: oder DEVS:clipboards.
MountList	Enthält die Beschreibungen der Geräte, die angemeldet werden können.
narrator.device	Steuert den Zugriff auf den Sprachsynthesizer.
parallel.device	Steuert den Zugriff auf den parallelen Anschluß.
printer.device	Steuert den Zugriff auf den Drucker.
ramdrive.device	Steuert den Zugriff auf die resetfeste RAM-Disk.
serial.device	Steuert den Zugriff auf den seriellen Anschluß.
system-configuration	Enthält die aktuellen Einstellungen des Voreinstellerprogramms Preferences.
Unterverzeichnis Keymaps	Enthält die Datei für die Tastaturbelegung d
Unterverzeichnis Printer	Enthält den Generic-Druckertreiber.
Unterverzeichnis Clipboard	Falls CLIPS: nicht zugewiesen ist, werden Zwischenspeicherinhalte hier gespeichert.

Verzeichnis L

Aux-Handler	Ermöglicht ungepufferte serielle Ein- und Ausgabe.
Disk-Validator	Erkärt Disketten gültig für das Schreiben, wenn sie in ein Laufwerk eingelegt werden.
FastFileSystem	Ein Dateisystem für den Einsatz mit Festplattenlaufwerken.
Newcon-Handler	Bietet ein interaktives SHELL-Fenster.
Pipe-Handler	Ermöglicht den Aufbau eines Kommunikationskanals zwischen ablaufenden Programmen.
Port-Handler	Steuert die DOS-Schnittstellen für PAR:, SER: und PRT:.
RAM-Handler	Ermöglicht die Speicherung von Dateien im Hauptspeicher.
SHELL-SEG	Steuert die SHELL.
Speak-Handler	Ermöglicht die Sprachausgabe des Amiga.

Verzeichnis Libs

diskfont.library	Enthält Funktionen für das Finden und Laden residenter Zeichensätze.
icon.library	Enthält Funktionen für das Erstellen und Lesen von Piktogramm-Dateien und Routinen für die Manipulation von Zeichenfolgen.
info.library	Enthält den Code für die Info-Funktion des Workbench-Menüs.
mathieeedoubbas.library	Enthält grundlegende IEEE-Rechenfunktionen mit doppelter Genauigkeit (Addition, Subtraktion, etc.).
mathieeedoubtrans.library	Enthält transzendentale IEEE-Rechenfunktionen mit doppelter Genauigkeit.
mathtrans.library	Enthält transzendentale Gleitkomma-Kurvenfunktionen (Sinus, Cosinus, etc.)
translator.library	Enthält eine Routine für Umsetzung von englischem Text in Phoneme, die für das Gerät "Narrator" geeignet sind.
version.library	Zeigt die Versions- und Revisionsnummern der Workbench-Ausgabe an.

Bitte beachten Sie, daß die Dateien mit ".info" in dieser Liste nicht aufgeführt sind. Sie enthalten die Informationen, die für die Anzeige der Piktogramme und Fenster und für den Start von Anwendungsprogrammen erforderlich sind. Die meisten Benutzer müssen nicht auf diese Dateien zugreifen.

C. Liste der Verzeichnisse/Dateien auf der Extras-Diskette

Die Dateien in den Verzeichnissen FD1.3 und Basic Demos beziehen sich auf die Programmiersprache Amiga-Basic. Weitere Informationen über diese Dateien entnehmen Sie bitte dem Amiga-BASIC-Handbuch, das mit dem Amiga geliefert wurde. Alle anderen Verzeichnisse auf der Diskette Extras werden nachstehend aufgeführt.

Verzeichnis Devs — Unterverzeichnis Keymaps

Tastenbelegung	*Zugehörige Tastatur*
cdn	Französisch-Kanadische Tastatur
ch1	Französisch-Schweizer-Tastatur
ch2	Deutsch-Schweizer Tastatur
d	Deutsche Tastatur
dk	Dänische Tastatur
e	Spanische Tastatur
f	Französische Tastatur
gb	Britische Tastatur
i	Italienische Tastatur
is	Isländische Tastatur
n	Norwegische Tastatur
s	Schwedische Tastatur
usa2	Dvorak-Tastatur

Verzeichnis Devs -- Unterverzeichnis Printers

Druckertreiber	*Kompatible Drucker*
Alphacom_Alphapro_101	Alphacom Alphapro 101
Brother_HR-15XL	Brother HR-15XL
Calcomp_ColorMaster	ColorView-5912
Calcomp_ColorMaster2	ColorView-5912
Canon_PJ-1080A	Canon PJ-1080A
CBM_MPS1000	CBM MPS1000, Canon BJ-130
Diablo_630	Diablo 630
Diablo_Advantage_D25	Diablo Advantage D25
Diablo_C-150	Diablo C-150
EpsonQ	Serie Epson Q (LS1500, LQ2500, etc.)
EpsonX	Serie Epson X (EX, FX, JX, LX. MX. RX, etc.), Commodore MPS1250
EpsonXOld	Serie Epson X, Star Micronics Gemini 10-X
Howtek_Pixelmaster	Howtek Pixelmaster
HP_DeskJet	Hewlett-Packard DeskJet
HP_LaserJet	Hewlett-Packard LaserJet, LaserJet PLUS und LaserJet II
HP_PaintJet	Hewlett-Packard PaintJet
HP_ThinkJet	Hewlett-Packard ThinkJet
Imagewriter II	Imagewriter, Imagewriter II
Nec_Pinwriter	NEC Pinwriter P5, P6, P7, P9, P2200
Okidata_92	Okidata 92
Okidata_293I	Okidata 293I
Okimate_20	Okimate 20 (mit seriellem und parallelem Schnittstellensatz "Plug'n Print")
Quadram_QuadJet	Quadram QuadJet
Qume_LetterPro_20	Qume LetterPro 20
Toshiba_P351C	Toshiba P351C, andere Toshiba-Drucker mit 24 Nadeln.
Toshiba_P351SX	Toshiba P351SX, andere Toshiba-Drucker mit 24 Nadeln.

Verzeichnis Tools

Fed	Ändert und erstellt Zeichensätze.
FreeMap	Zeigt die Auslastung des CHIP-Speichers an.
IconEd	Ändert die Bilddarstellung von Piktogrammen.
IconMerge	Mischt oder trennt Doppel-Piktogramme.
KeyToy2000	Zeigt die gesamte Tastaturbelegung eines Amiga 2000/500.
MEmacs	Editor für Textdateien.
Palette	Ändert die Farben eines Bildschirms.
PerfMon	Zeigt die Leistungsauslastung des Systems.

Verzeichnis Fonts

Zeichensätze	*Schriftgrößen*
Courier	11, 13, 15, 18, 24
Helvetica	9, 11, 13, 15, 18, 24
Times	11, 13, 15, 18, 24

Bitte beachten Sie, daß die Dateien mit ".info" in dieser Liste nicht aufgeführt sind. Sie enthalten die Informationen, die für die Anzeige der Piktogramme und Fenster und für den Start von Anwendungsprogrammen erforderlich sind. Die meisten Benutzer müssen nicht auf diese Dateien zugreifen.

Commodore